UNIVERSITE MOHAMMED V AGDAL

ECOLE MOHAMMADIA D'INGENIEURS



Filière: Informatique

Option: Système d'information

Mémoire de Projet de Fin d'Etudes

N° 2012/40

ETUDE ET REALISATION D'UNE SOLUTION D'INTEGRATION D'UN SYSTEME GED ET UN SI SOUS L'ARCHITECTURE SOA

Réalisé par :

Mlle Oubel Meriem

Dirigé par :

M. Khalidi Idrissi Mohammed

M. El Boudihi Iliass

Année : 2011/2012

UNIVERSITE MOHAMMED V AGDAL
ECOLE MOHAMMADIA D'INGENIEURS
FILIERE GENIE INFORMATIQUE



EMI / GINFO / 2012-40

Projet de Fin d'Etudes

Présenté par

Oubel Meriem

Pour l'obtention du Diplôme d'Ingénieur d'Etat en Informatique

Etude et réalisation d'une solution d'intégration d'un système GED et un SI sous l'architecture SOA

Soutenu le 31/05/2012

Devant le Jury composé de :

Mme. F.Belouadha	**Professeur à l'EMI**	**Présidente**
Mme. A. Retbi	**Professeur à l'EMI**	**Rapporteur**
M. I. El Boudihi	**Architecte chef de projet**	**Encadrant**
M. M. KHALIDI IDRISSI	**Professeur à l'EMI**	**Encadrant**

Année Universitaire 2011/2012

باسم الله الرحمان الرحيم

ٱقۡرَأۡ بِٱسۡمِ رَبِّكَ ٱلَّذِى خَلَقَ ١

خَلَقَ ٱلۡإِنسَـٰنَ مِنۡ عَلَقٍ ٢ ٱقۡرَأۡ

وَرَبُّكَ ٱلۡأَكۡرَمُ ٣ ٱلَّذِى عَلَّمَ

بِٱلۡقَلَمِ ٤ عَلَّمَ ٱلۡإِنسَـٰنَ مَا لَمۡ

يَعۡلَمۡ ٥

صدق الله العظيم

Dédicace :

A ma chère mère,

A mon cher père,

A mon cher frère,

A mes chères sœurs,

Nul mot ne pourra exprimer mes sentiments et ma gratitude envers vous,

A toute la famille

A mes chers amis

A tous ceux qui m'aiment,

A tous ceux que j'aime,

Je dédie ce travail…

Meriem Oubel

Remerciements

Au terme de ce stage, je tiens à remercier tout le personnel de la société Créative Technologies et spécialement l'équipe de l'atelier du développement avec qui j'ai passé ma période de stage.

Je témoigne de toute ma reconnaissance à Monsieur Iliass El Boudihi architecte et responsable recherche et développement, pour m'avoir encadré et encouragé tout au long de ce projet, et prodigué ses directives précieuses et ses conseils pertinents qui m'ont été d'un appui considérable dans ma démarche.

Mes remerciements les plus sincères à mon encadrant M. Mohammed KHALIDI IDRISSI, qui, grâce à ses conseils constructifs et ses critiques pertinentes, m'a permis de me remettre en question et donner le meilleur de moi-même.

Je tiens aussi à remercier tous les professeurs membres du jury qui m'ont fait l'honneur d'accepter de juger mon travail, en particulier Mme Fatimazohra Belouadha et Mme Asmaa Retbi.

Remerciements spéciaux à tout le corps professoral de l'EMI, pour la formation de qualité qu'il m'a prodiguée.

Que tous ceux et celles qui ont contribué de près ou de loin à l'accomplissement de ce travail trouvent l'expression de mes remerciements les plus chaleureux.

Résumé

La gestion électronique des documents (GED) désigne un procédé informatisé visant à organiser et gérer des informations et des documents électroniques au sein d'une organisation. Il met principalement en œuvre des systèmes d'acquisition, de classement, de stockage, d'indexation et de consultation. La GED participe aux processus de travail collaboratif, de capitalisation et d'échanges d'informations. Elle peut permettre un gain de temps et d'argent énorme pour les organisations.

Diverses stratégies permettent de parvenir à l'intégration des systèmes GED avec les systèmes d'information, et chacune affiche des atouts spécifiques. Elles permettent de récupérer la donnée brute et la diffuser et l'exposer au service concerné, elles rencontrent de plus en plus de succès auprès des entreprises.

Mon projet de fin d'études s'inscrit dans cette optique, en mettant en place une démarche d'installation et de paramétrage du système GED choisi, il s'agit ensuite de l'intégrer avec les systèmes de l'entreprise afin d'acquérir les informations brutes et éparses et mettre à leur disposition la donnée de référence. En adoptant une architecture orientée service, il est question de trouver des solutions connectant le logiciel de gestion des documents avec le SI pour qu'ils puissent communiquer les documents et les données entre eux.

Abstract :

The electronic document management (EDM) is a computerized method to organize and manage information and electronic documents within an organization. It implements an acquisition, classification, storage and indexing systems. EDM participates in collaborative work, funding and information exchange processes. It can help save time and money for organizations.

Various strategies are used to achieve GED system integration with information systems, and each shows specific strengths. They allow recovering the raw data and exposing it to the concerned service, they encounter increasingly popular with businesses.

My project of studies end is a part of this perspective. It’s setting up a process of installation and setup of chosen EDM system, the next step is to integrate this latter with enterprise systems in order to acquire the raw and scattered information, and provides them with the reference data. By adopting service oriented architecture, it comes to find solutions connecting the document management software with the IS so that they can communicate the documents and data between them.

ملخص

إدارة الوثـــائق الإلكترونيـــة هـــو نهـــج يعتمـــد علـــى تقنيـــات المعلوميـــات لتنظـــيم وإدارة المعلومـــات والوثـــائق الالكترونيـــة داخـــل المؤسســـة. هـــذا البرنـــامج يمـــنح أنظمـــة إقتنـــاء و ترتيـــب و تخـــزين و فهرســـة و بحـــث. يســـاهم فـــي العمـــل المشـــترك و فـــي التبـــادل المعلومـــاتي. و يعتبـــر صـــفقة مربحـــة بالنســـبة للشـــركة المتبنيـــة مـــن حيـــث تـــوفير الوقـــت و المال.

تتعـــدد الاســـتراتيجيات التـــي تتـــيح ربـــط برنـــامج إدارة الوثـــائق بالنظـــام المعلومـــاتي للشـــركة. إذ يعمـــل علـــى اســـتقطاب المعلومـــة الخـــام و مـــن ثـــم نقلهـــا إلـــى القســـم المعـــين. هـــذا النظام يعرف إقبالاً كبيراً من طرف الشركات

و عليـــه ينـــدمج مشـــروع نهايـــة الدراســـة هـــذا فـــي نفـــس الإطـــار، و ذلـــك بتثبيـــت برنـــامج لإدارة الوثـــائق الإلكترونيـــة و تحديـــد معـــاييره و ربطـــه بالنظـــام المعلومـــاتي للشـــركة و دلك باعتماد أسلوب الخدمات

Liste des figures

Liste des tableaux :

Table des matières :



Introduction générale :

La recherche d'un document s'avère plus coûteuse que sa production, ce qui rend tentant de numériser un document et ne traiter que sa version digitale, afin de réduire les difficultés de recherche, diminuer les ressources mobilisées et assurer à l'entreprise un système d'information plus performant et plus efficace.

Un système GED: Gestion Electronique des Documents, désigne le processus de gestion du cycle de vie d'un document électronique, de sa création à sa destruction. Il vient pour répondre à ces fins ; son objectif est de parvenir à constituer un coffre-fort électronique de l'ensemble des documents pertinents pour l'entreprise et ce, pour atteindre un objectif intégral : optimiser ses processus métiers en traitant une seule fois l'information et en la mettant à disposition au bon moment aux bonnes personnes.

Les données numérisées et archivées dans la GED doivent être transmises au système d'information de l'entreprise afin qu'elles subissent le traitement nécessaire et atteignent leur destination. Une fois arrivée au système d'information, les données intègrent le Workflow ou sont directement envoyées vers l'entité concernée.

C'est dans ce cadre que s'inscrit mon projet de fin d'étude ayant comme objectif l'offre d'une solution d'intégration d'un système GED dans un SI, afin d'assurer la transmission des données par un connecteur, après avoir installé et paramétré la GED choisie.

Le présent rapport décrit le travail réalisé. Dans un souci de clarté il est décomposé en 6 chapitres. Le premier chapitre présente le contexte général du projet ainsi que l'organisme d'accueil, dans le deuxième chapitre un benchmark des systèmes GED existants sera effectué afin de choisir le système à installer, dans un troisième chapitre nous présenterons l'architecture orientée service ainsi que la première solution basée sur les Web service. Le chapitre suivant détaillera d'autres solutions d'intégration qui sont le MOM, l'EAI et l'ESB. Finalement le dernier chapitre présentera une étude comparative des connecteurs et tranchera de la solution à adopter.



Chapitre 1 : Contexte Général du projet

Le présent chapitre aborde le cadre général du projet, il présente d’abord l’organisme d’accueil, ensuite il étale la méthode et choix qui ont imprégnés la conduite de ce projet.



1.1. Présentation de l'organisme d'accueil

Creative technologies est une SSII marocaine fondée en 2000, filiale du groupe CDG Développement depuis 2009, elle concrétise ses ambitions d'accompagner ses clients dans l'intégration de solutions informatiques.

La première décennie dans la vie d'une SSII est très significative. Pendant cette période, Creative Technologies a fait valoir sa force et sa jeunesse pour s'installer fortement sur des secteurs clés des technologies de l'information.

L'organisation de Creative Technologies reflète sa méthodologie de travail et son but d'exceller et de réussir ses projets. Pour cela, ils adoptent une gestion par projet et dédient chaque catégorie de projets à une unité, la Business Unit. Et en support aux projets, viennent les directions Usine logicielle, Architecture et Recherche et Développement, Commerciale et Vente et la Direction Finances et support, pour compléter leur organisation transversale et afin d'assurer une meilleure production et qualité.

Creative offre à ses clients une panoplie de services :

- ✓ **Développement des ERP** qui est un outil incontournable pour rationaliser les processus de l'entreprise et contribuer ainsi au décloisonnement des fonctions en facilitant l'échange de données. L'ERP contient en son sein une part assez importante des données essentielles à l'information des décideurs. Grâce à une méthodologie éprouvée de gestion de projets, Creative Technologies apporte une expertise de pointe dans la mise en œuvre de progiciels.

- ✓ Dans sa composante **conseil autour des ERP**, Creative Technologies intervient sur la vision des évolutions organisationnelles associées à l'implémentation d'un progiciel intégré et sur les réflexions en amont qui impactent le nouveau référentiel de données induit par le choix de l'ERP.

- ✓ **BI : Business Intelligence** L'augmentation du nombre d'applications ERP, CRM, systèmes de Business Intelligence, outils de supervision des systèmes, applications internet et extranet, applications verticales (RH, finances, achats) multiplie les échanges entre toutes ces applications. Pour améliorer la diffusion des informations



et permettre de meilleures prises de décision, l'entreprise doit disposer d'un SI temps réel mettant en œuvre des indicateurs de pilotage. Creative Technologies accompagnent ses clients dans la conception et la réalisation des tableaux de bord qui leur permettent d'avoir une vision complète et pertinente de leurs activités.

- ✓ **Système d'Information Technique et de Gestion :** La mise à niveau d'une société passera nécessairement par une utilisation efficiente de l'outil informatique, par la mise en œuvre de systèmes d'information adaptés et performants. Creative Technologies met toute son expertise au service des administrations, offices publics et entreprises souhaitant des solutions spécifiques.

Creative offre d'autres services à savoir la revente des licences, Telecom et convergence IP, elle met à la disposition de son client les outils pour parfaire la maîtrise en intégrant l'ensemble de ses préoccupations:

- Faire dialoguer l'ensemble de ces systèmes pour réaliser un système complètement intégré.
- Systèmes interopérables.
- Portail décisionnel et opérationnel intégrant les informations des bases de données géo-spatiales et documentaires.
- Architecture Intranet/Extranet/Internet.
- Systèmes embarqués.

1.2. Présentation générale du projet

Chaque entreprise a besoin d'un système qui s'occupera de la gestion de ses documents, une gestion électronique s'avère la plus bénéfique. La vocation de l'automatisation peut être administrative, documentaire, technique, bureautique ou d'archivage, le document dématérialisé devient donc porteur de connaissance et d'efficacité partagée au sein des équipes de travail. Aussi des retombées économiques sont également attendues de ce système par la qualité et la rapidité du traitement des demandes des partenaires ou collaborateurs de l'entreprise.



En adoptant un système GED dans l'entreprise, il s'avère incontournable de le connecter avec le système d'information. Cette intégration permet de fournir le document nécessaire au moment convenable au SI. La GED constitue donc une brique du système globale de l'entreprise où elle sert de stockage centralisé pour l'ensemble des fichiers de l'organisation.

La question se pose au niveau de la technique d'intégration, le projet consiste à trouver une solution qui interface un système GED avec un système d'information quelconque, il s'agit donc de connecter la GED au SI de l'entreprise en adoptant une solution optimale qui répond aux besoins fonctionnels.

1.3 Conduite du projet

1.3.1 La méthode Scrum

La nature du projet incite à suivre une méthode agile et plus précisément la méthode Scrum. En effet les spécifications du projet non pas cessé d'être modifiés tout au long du projet et les outils et les technologies n'ont pas été explicites depuis le départ. Scrum permet d'entamer le projet avec un minimum d'information sur les travaux à faire et qui sont ajustés régulièrement au cours du projet. Le cycle de vie de la méthode Scrum se décompose en plusieurs Sprint successifs :

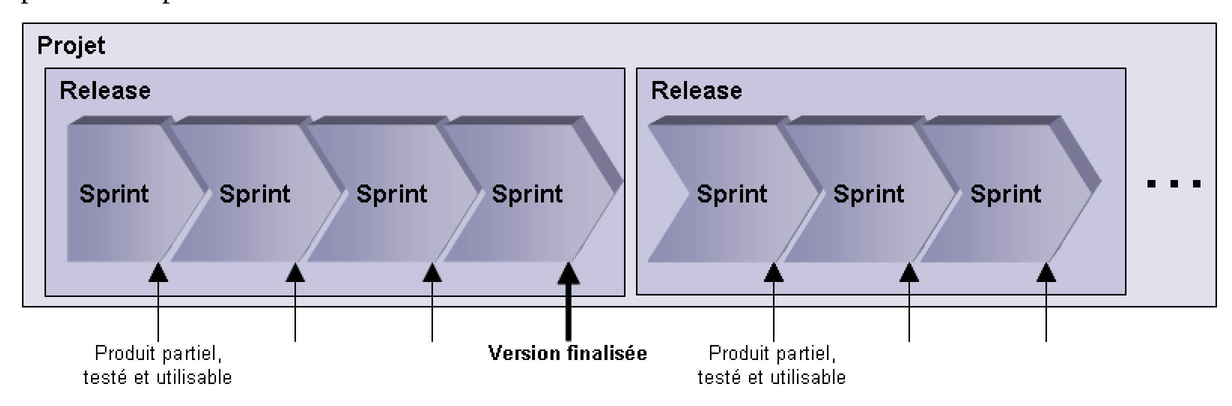


Figure 1.1 Projet Scrum [1]

Le Sprint dure au maximum 4 semaines, et pendant ce cycle des réunions quotidiennes sont menées pour faire le point sur le travail accompli et les obstacles rencontrés depuis la dernière réunion. Des livraisons fréquentes sont attendues à la fin du Sprint. Le client fait partie intégrante du projet.



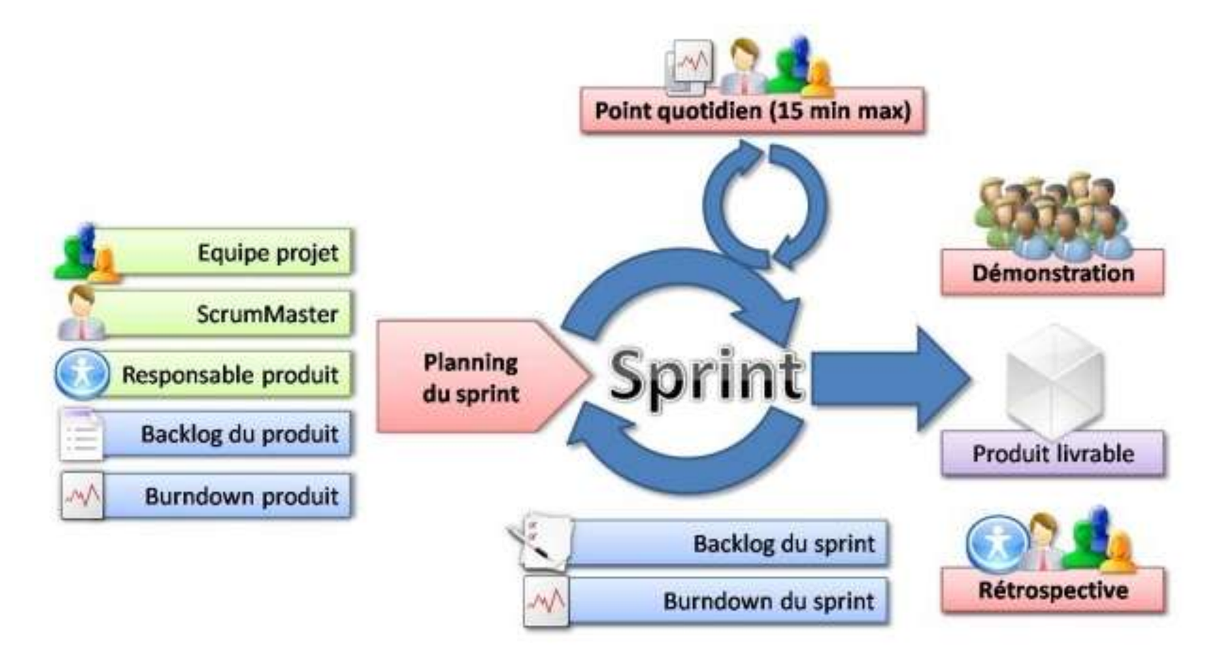


Figure 1.2 Cycle Scrum [2]

Le ScrumMaster est le responsable de la méthode. Il doit s'assurer que celle-ci est comprise, et bien mise en application. Ce n'est pas un chef de projet, ni un intermédiaire de communication avec les clients.

Le responsable produit est dans ce cas le chef de projet encadrant de l'équipe du projet.

Le Backlog du produit est la liste des fonctionnalités attendues d'un produit. Il est présenté en tant que livrable du produit installé ou cahier de charge de l'étude de la solution d'intégration effectuée.

Le Burndown est une sorte de graphique d'avancement, il montre la tendance du reste à faire total de jour en jour, il est présenté sous forme de bilan du jour pour le suivi du projet.

1.3.2 Diagramme de Gantt :

Le diagramme de GANTT est un outil permettant de modéliser la planification de tâches nécessaires à la réalisation du projet, il visualise dans le temps les diverses tâches liées composant le projet en fournissant une représentation graphique de son avancement.

Pour tracer le diagramme de Gantt nous avons utilisé MS Project, et voici le tableau citant les tâches du projet.



Nom de la tâche	Durée	Début	Fin	Prédécesseurs
Benchmark des GED	3 jours	Mer 01/02/12	Ven 03/02/12	
Documentation eDOCS	3 jours	Lun 06/02/12	Mer 08/02/12	1
Installation eDOCS	6 jours	Jeu 09/02/12	Jeu 16/02/12	2
Documentation CLM	5 jours	Ven 17/02/12	Jeu 23/02/12	1
Installation CLM	10 jours	Ven 24/02/12	Jeu 08/03/12	4
Paramètrage CLM	10 jours	Ven 09/03/12	Jeu 22/03/12	5
Exploitation CLM	40 jours	Ven 23/03/12	Jeu 17/05/12	6
Etude solution web service	9 jours	Mar 07/02/12	Ven 17/02/12	
Etude du protocole de communication REST & SOAP	7 jours	Lun 20/02/12	Mar 28/02/12	8
Cahier de charge interface REST	8 jours	Mer 29/02/12	Ven 09/03/12	9
Etude de l'architecture MOM	10 jours	Lun 12/03/12	Ven 23/03/12	10
Cahier de charge MOM	10 jours	Lun 26/03/12	Ven 06/04/12	11
Etude de la solution EAI	3 jours	Lun 09/04/12	Mer 11/04/12	12
Etude de la solution ESB	5 jours	Jeu 12/04/12	Mer 18/04/12	13
Implémentation de l'architecture MOM	22 jours	Jeu 19/04/12	Ven 18/05/12	14
Rédaction du rapport	75 jours	Lun 06/02/12	Ven 18/05/12	

Tableau 1.1 : Les tâches du projet



23 Jan 12 | 06 Fév 12 | 20 Fév 12 | 05 Mar 12 | 19 Mar 12 | 02 Avr 12 | 16 Avr 12 | 30 Avr 12 | 14 Mai 12 | 28 Mai 12

Sprint 1: Benchmark GED, Documentation eDOCS, Installation eDOCS

Sprint 2: Documentation CLM, Installation CLM, Paramétrage CLM

Sprint 3: Exploitation CLM

Sprint 4: Etude Web Service, Etude REST vs SOAP, Cahier de charge REST

Sprint 5: Etude MOM, Cahier de charge MOM

Sprint 6: Etude EAI, Etude ESB, Implémentation MOM

Rapport

Figure 1.3 : Diagramme de Gantt

1.3. Conclusion

Dans ce chapitre, nous avons présenté le contexte général du projet, en introduisant l'organisme hôte Creative Technologies.

Ensuite nous avons détaillé la conduite du projet en précisant la méthode suivie, et en citant la liste des tâches réalisées illustrée par le diagramme de Gantt.



Chapitre 2 : Cahier de charge

Dans ce chapitre une étude de l'existant sera menée afin d'assurer une meilleure compréhension du système cible. Aussi les objectifs et les enjeux seront- ils étalés pour modéliser le système à étudier.



La mission de diagnostic de l'existant et de cadrage fonctionnel revêt une importance capitale compte tenu des objectifs à atteindre par le projet à réaliser. Elle constitue le fondement de la suite du projet en fournissant tous les éléments et contraintes à prendre en compte tout au long du déroulement du projet.

2.1 Etude de l'existant :

L'objet de ce paragraphe est d'énoncer les spécifications pour l'analyse fonctionnelle. Il s'agit de ressortir les entités et concepts fondamentaux du domaine à informatiser ainsi que les relations qui existent entre eux. Ces spécifications sont déduites de l'organisation existante et de l'ensemble des ateliers de travail avec l'ensemble des interlocuteurs.

2.1.1 Entretiens pour la collecte de l'information

Les documents circulant au sein de Creative sont considérés comme étant des courriers. Ils passent par un Workflow qui commence souvent du bureau d'ordre, s'acheminent vers les différents services concernées pour tout traitement approprié et s'achèvent par le stockage. Et pour bien cerner les circuits des différents types des courriers, des séances de travail avec les principaux intervenants dans la circulation du courrier ont été menées.

Une réunion a été réalisée sur la base d'atelier de travail dans le but de recueillir les informations de nature quantitative et qualitative (volumes stockés dans les bureaux, accroissement journalier, descriptif des dossiers types, etc.) avec l'assistante du bureau d'ordre, ces ateliers nous ont permis d'identifier :

- Les outils de traitement du courrier: Registre d'enregistrement, critères de classement et de recherche.
- Les méthodes et moyens de classement et stockage des courriers.
- Les types des courriers et leurs catégories :

Type de courrier	**Courrier entrant et sortant**
Catégorie du courrier arrivé	**Fax, bordereau d'arrivé**
Nature du courrier arrivé	Décompte, Facture , Relevée bancaire, Demande d'emploi, PV de réunion signé, Compte rendu de réunion, marché, Ordre de service, Rapport d'étude, Bon de livraison, bon de commande,

	Lettre d'éclaircissement, avis d'appel d'offres , Convention, Devis, Relevé bancaire, Appel d'offres etc.
Catégorie du courrier départ	Fax, Bordereau d'envoi.
Nature du courrier départ	Bordereau d'envoi, bon de commande, devis, réponse d'appel d'offre, facture, offre de prix, caution, demande de caution, restitution de caution, rapport de cadrage, bon de restitution, bon de mise à disposition, déclaration fiscale, etc.

Tableau 2.1 : Les catégories du courrier

2.1.2 Cycle d'un courrier entrant

Propositions pour le Courrier Entrant

Expéditeur
Bureau d'Ordre Central (BOC)
Assistance Direction Générale (SDG)
Directeur Général(DG)
Direction(D)
Business Unit(BU)

Envoi du courrier
Réception et tri du courrier
Numérisation du courrier
Pré-enregistrement du courrier
Transfert du courrier
Réception du courrier
Validation de l'enregistrement
Transfert du courrier
Annulation du Transfert
Traitement du courrier Préparation d'un courrier départ
Classement physique du courrier au niveau du BOC selon les modalités définies
Acceptation du transfert
Rejet ou Annulation
Acceptation
Transfert du courrier
Annulation du Transfert
Acceptation et transfert
Acceptation
Traitement du courrier Affectation du courrier Préparation d'un courrier départ
Rejet ou Annulation
Archivage du fond du dossier
Annulation du Transfert
Acceptation et transfert
Acceptation
Traitement du courrier Affectation du courrier Préparation d'un courrier départ
Rejet ou Annulation
Archivage du fond du dossier

Figure 2.1 : Diagramme d'activité du cycle d'un courrier



Ainsi le courrier une fois arrivé au bureau d'ordre, il est transmis vers les services concernés pour qu'il soit traité et renvoyé ensuite vers l'assistante de direction afin que le directeur le valide. Après validation le courrier est remis au bureau d'ordre et est stocké.

Cependant, Le cycle de vie du document peut être très long quand il s'agit d'un document qui doit passer par plusieurs services. Le fait de faire circuler un papier manuellement provoque un gaspillage de temps par rapport au même cycle automatisé. Le document électronique peut être envoyé dès la terminaison de son traitement au service qui suit, ce qui évite le temps d'attente de récupération du document.

Un système de gestion électronique des documents automatisera donc le cycle du courrier et constituera un volet transversal pour l'ensemble des entités suivantes: Réception, Enregistrement et référencement, dispatching du courrier entrant et envoi du courrier sortant vers les différents destinataires externes (Groupe CDG, Filiales, Administration Publique, Ministères, Wilaya, Agence Urbaine, Fournisseurs, Partenaires, Office, etc.), et échange de courrier au sein même de Creative Technologies.

2.2 Cahier de charge fonctionnel

La société Creative Technologies a décidé de mettre en place un système de gestion de courrier basé sur un système de gestion électronique des documents (GED). Cette initiative automatise le cycle de vie d'un document circulant au sein de l'entreprise et permet de bénéficier de tous les avantages d'un système de gestion électronique de document.

2.2.1 Les objectifs

Les objectifs et l'approche globale du projet montrent que le projet est né d'un besoin réel et d'une stratégie de modernisation de la circulation du courrier au niveau de Creative Technologies.

Ce projet a deux objectifs principaux :

- L'installation d'une GED qu'il s'agit de déterminer, tout en fournissant un guide détaillé de l'installation et un manuel d'utilisation.
- L'établissement de la connexion entre le système GED choisi et le SI.

Pour la phase d'installation il s'agit de mettre en place un système qui s'occupera de la gestion des documents de Creative en automatisant les flux de données par la réalisation des Workflows et le stockage électronique de ces documents. Le système implémenté doit donc



être paramétré afin qu'il prend en compte tous les processus des courriers de la société. Les livrables de cette partie seront présentés sous forme de guide d'installation et de manuel d'utilisation.

Pour la connexion entre la GED et le SI, il s'agit de fournir plusieurs solutions et établir une étude comparative pour choisir par la suite la plus adéquate aux besoins fonctionnels de l'entreprise. Et le livrable de l'étude se manifeste en un cahier de charge et un dossier de spécification détaillé du connecteur choisi.

2.2.2 Les besoins fonctionnels :

La volonté des responsables est de mettre en œuvre progressivement un système performant homogène, simple d'utilisation et adapté à l'ensemble des entités. Et de l'intégrer au SI afin qu'ils communiquent entre eux les documents et les informations nécessaires.

Les objectifs assignés au projet s'articulent autour des axes suivants :

- Assurer une traçabilité et un suivi du courrier,
- Mettre en œuvre les moyens pour anticiper au lieu de subir,
- Favoriser l'efficacité administrative et la facilité de prise de décision,
- Améliorer la réactivité de l'entreprise ainsi que les délais de réponse.

Afin d'atteindre ces objectifs des actions sont menées en parallèle de la réalisation du projet. Pour éviter la résistance du personnel, nous avons accompagné l'ensemble des structures au changement induit par l'introduction de la nouvelle technologie. Aussi de temps en temps nous organisons des présentations pour les chefs de projets afin qu'ils donnent leurs avis de la solution trouvée et que nous puissions collecter les besoins qui apparaissent lors du déroulement du projet.

2.2.3 Les enjeux :

Après l'installation de la GED il s'agit de la connecter au/aux SI de l'entreprise. Il faut donc chercher une solution d'intégration qui assure l'ensemble des fonctionnalités suivantes :

- Un couplage faible : ou couplage lâche en effet les applications vont échanger des données et le connecteur jouera le rôle de chef d'orchestre pour l'ensemble des échanges d'informations, il permettra de passer d'un couplage fort en cas d'absence de solution d'intégration à un couplage faible entre les applications comme l'illustre le schéma suivant :



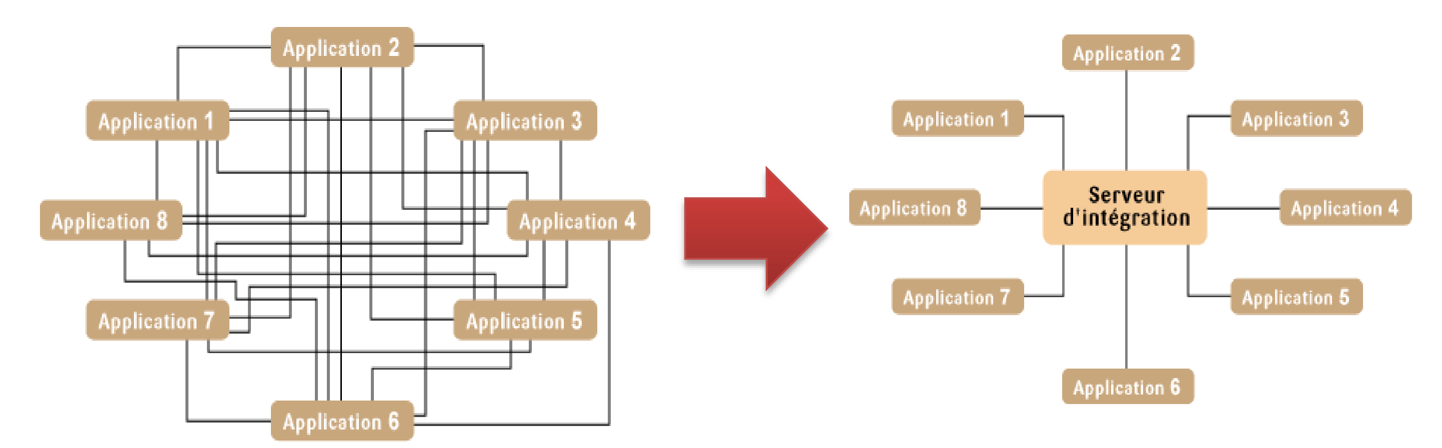


Figure 2.2 : Couplage lâche [3]

- Communication asynchrone : Le client et le serveur peuvent ne pas être connectés en même temps et l'appel de procédure à distance fonctionne sans problème. Le client envoie donc les données vers le connecteur sans se soucier de l'état du destinataire et c'est le connecteur qui s'occupera du transfert des données quand le destinataire sera disponible.
- Intégrer des systèmes hétérogènes : Les systèmes à intégrer ne sont pas nécessairement bâtis sur des technologies semblables, il s'agit donc de connecter des applications de différents langages de programmation, le connecteur doit ainsi être indépendant de la technologie des clients qu'il reliera.
- Solution scalable et configurable : Il faut prévoir un gros volume de données à transférer et des communications multipoints, c'est-à-dire relier P clients consommateurs avec N clients producteurs, et assurer donc des échanges asymétriques.
- Solution évolutive : La solution à implémenter doit avoir une communauté d'utilisateurs active.
- Mise en œuvre de la solution : il faut maîtriser les coûts de réalisation ou d'obtention de la solution, ainsi que le temps d'implémentation.

2.3 Conclusion :

Ce chapitre a mis en exergue les objectifs qui ont orienté le travail avant de détailler les enjeux du projet. Une étude de l'existant a été présentée afin de mieux comprendre la nécessité du système à déployer. Dans la suite nous présenterons le système GED qui sera installé.



Chapitre 3 : Choix de la GED

Ce chapitre présente un benchmark des systèmes GED et précise la GED choisie afin qu'elle soit installée et connectée avec le SI.



3.1 La gestion électronique des documents :

La gestion du cycle de vie du document recouvre un ensemble d'activités : capture, création, gestion, révision, distribution, publication, stockage et conservation. Ces activités appliquées systématiquement au contenu de la structure suivant des règles définies par l'organisation elle-même ou prescrites par son environnement réglementaire.



Figure 3.1 Processus métier de la GED [4]

Les apports de la GED sont très nombreux, outre les gains en termes d'organisation, de suivi et d'information. Mais il existe d'autres qui sont directement chiffrables :

- Coût de l'espace de stockage
- Coûts de classement
- Temps de recherche
- Temps de diffusion
- Accès simultané
- Gestion des versions
- Sûreté et Sécurité

Une fois le besoin exprimé, il est temps de s'intéresser aux différents acteurs du marché et de mener une étude comparative des solutions de GED. Un tableau synthétique comparatif des différentes solutions, orienté selon les besoins exprimés, peut aider à obtenir une vue d'ensemble de l'offre. Il s'agit alors de vérifier l'adéquation entre les différentes solutions possibles et le besoin exprimé.



3.2 Etude comparative

Afin de choisir un système de gestion électronique, il fallait mener une étude comparative des systèmes GED existants, pour cette fin nous avons cherché les systèmes les plus performants et connus et nous avons établi le tableau suivant, qui met l'accent sur les critères les plus parlants :

Logiciel	Prix	Points forts	Point faible	Fonctionnalité	OS compatibles
Alfresco	Abonnement obligatoire à partir de 16 125 euros par CPU (environ 50 utilisateurs)	Rapidité, robus tesse Etendue fonctionnelle	Complexe à configurer (fichiers XML)	Business Process/Workflow Gestion des emails Recherches fulltext Sauvegarde de recherches Gestion des droits Authentification LDAP/SSO Records management Archivage légal Web Content Management	Serveurs: Windows Server 2003, 2008 Red Hat Enterprise Linux 5 Sun Solaris 10 Clients: Windows 2000, XP,Vista Mac OS X
Nuxeo	Gratuit (licence LGPL)	Gratuité des licences, pas d'abonnement obligatoire Etendue fonctionnelle Flexibilité Capacités d'extension et d'intégration aux systèmes d'informations existants	Complexe à configurer	Business Process/Workflow Gestion des emails Recherches fulltext Sauvegarde de recherches Gestion des droits Authentification LDAP/SSO Records management Archivage légal Web Content Management	Serveurs: Windows Server 2000, 2003,2008 RedHat Linux 3.x, 4.x, 5.x Sun Solaris 9.x, 10.x Clients: Windows XP,Vista, Mac OS X



Sharepoint Editeur: Microsoft	License Serveur MOSS 2007: 3730€ Licence Client (CAL): 80€	Portail collaboratif performant et simple à utiliser Solution moins couteuse et moins complexe à mettre en œuvre que Opentext et Documentum	Fonction-nalités GED moins avancées que ses concur-rents, notam-ment en terme d'archivage légal et longue durée	Business Process/Workflow Gestion des emails Recherches fulltext Sauvegarde de recherches Gestion des droits Authentification LDAP/SSO Records management Web Content Management	Serveurs: Windows Server 2003, 2008 Clients: Windows XP, Vista Mac OS X
OpenText ECM Suite (ex Livelink) Editeur: OpenText	Prix minimum de départ non com-muniqué (Cout de 1000 licences:475 400 €)	Fonctionnalités GED optimisées grâce à l'intégration MS Office, Sharepoint, SAP, etc.	Difficulté d'accès pour les non-informaticiens	Business Process/Workflow Gestion des emails Recherches fulltext Sauvegarde de recherches Gestion des droits Authentification LDAP/SSO Records management Archivage légal Web Content Management	Serveurs: Windows Server 2003, 2008 Red Hat Enterprise Linux 5 Sun Solaris 10 Clients: Windows 2000, XP, Vista Mac OS X
EMC Documentum Editeur: EMC	Prix minimum de départ non com-muniqué (Cout de 1000 licences: 557300 €)	Rapidité, robustesse: le temps de réponse moyen est de 0,86s pour 100000 utilisateurs connectés simultanément pendant 12H	GED la plus chère du marché, complexe et longue à mettre en œuvre	Business Process/Workflow Gestion des emails Recherches fulltext Sauvegarde de recherches Gestion des droits Authentification LDAP/SSO Records	Serveurs: Windows Server 2003, 2008 Red Hat Enterprise Linux 5 Sun Solaris 10 Clients: Windows 2000, XP, Vista Mac OS X



				management Archivage légal Web Content Management	

Tableau 3.1 : Etude comparative GED

Synthèse:

Le tableau donne une vision claire sur les champs permettant d'effectuer une comparaison entre les différents systèmes GED des différents éditeurs, en se basant sur le critère du point faible, la première GED éliminée est Sharepoint parce qu'elle présente des fonctionnalités moins avancées que ses concurrents au marché.

Alfresco met l'accent sur le travail collaboratif. Il s'intègre particulièrement bien à Microsoft Office et Microsoft Sharepoint. Il est en revanche complexe à installer et à configurer. Nuxeo est disponible gratuitement en version complète, il se caractérise par son étendue fonctionnelle, sa flexibilité, ses capacités d'extension et d'intégration aux systèmes d'informations existants mais il est aussi complexe à installer et à configurer.

La complexité de configuration est un critère éliminatoire des GED Alfresco et Nuxéo. En effet une GED qui ne peut pas être configurable selon les besoins de l'entreprise ou qui sa configuration est très coûteuse ne peut pas être une solution convenable à notre cas.

Pour EMC Documentum malgré sa rapidité et sa robustesse qui la distingue de ses concurrents, elle ne peut être privilégiée à cause de son prix, elle est en fait la plus cher du marché, elle est aussi complexe et nécessite une longue durée pour sa mise en œuvre.

Quant à OpenText, il propose des GED avec des prix convenables et des fonctionnalités avancées. Le seul inconvénient est leur difficulté d'utilisation pour les non-informaticiens. Mais ceci est un problème facilement réglable en organisant des séances de formation aux employés qui vont utiliser le logiciel. En conséquence, la GED choisie est le produit de l'éditeur OpenText.



3.3 Présentation de la GED choisie

3.3.1 L'éditeur Open Text

La GED choisie est l'un des produits de l'éditeur Open Text, un éditeur de logiciels, premier fournisseur indépendant de solutions de Gestion de Contenu pour l'Entreprise (ECM), solutions qui fédèrent les personnes, les processus et l'information. Les solutions d'Open Text allient Travail Collaboratif et Gestion de Contenu, Archivage transformant l'information en connaissance, ce qui est la base de l'innovation, du respect des normes en vigueur et de la croissance.

La richesse fonctionnelle des produits OpenText s'appuie sur une combinaison unique d'outils de gestion documentaire, travail collaboratif, de gestion des processus automatisés d'entreprise, de gestion de contenu web, de gestion de la connaissance de recherche et d'archivage. L'ensemble de l'offre est étroitement intégré dans une solution facile à personnaliser et déployer. Son positionnement de type progiciel permet de réduire drastiquement les ressources en déploiement des solutions et de réduire ainsi les coûts.

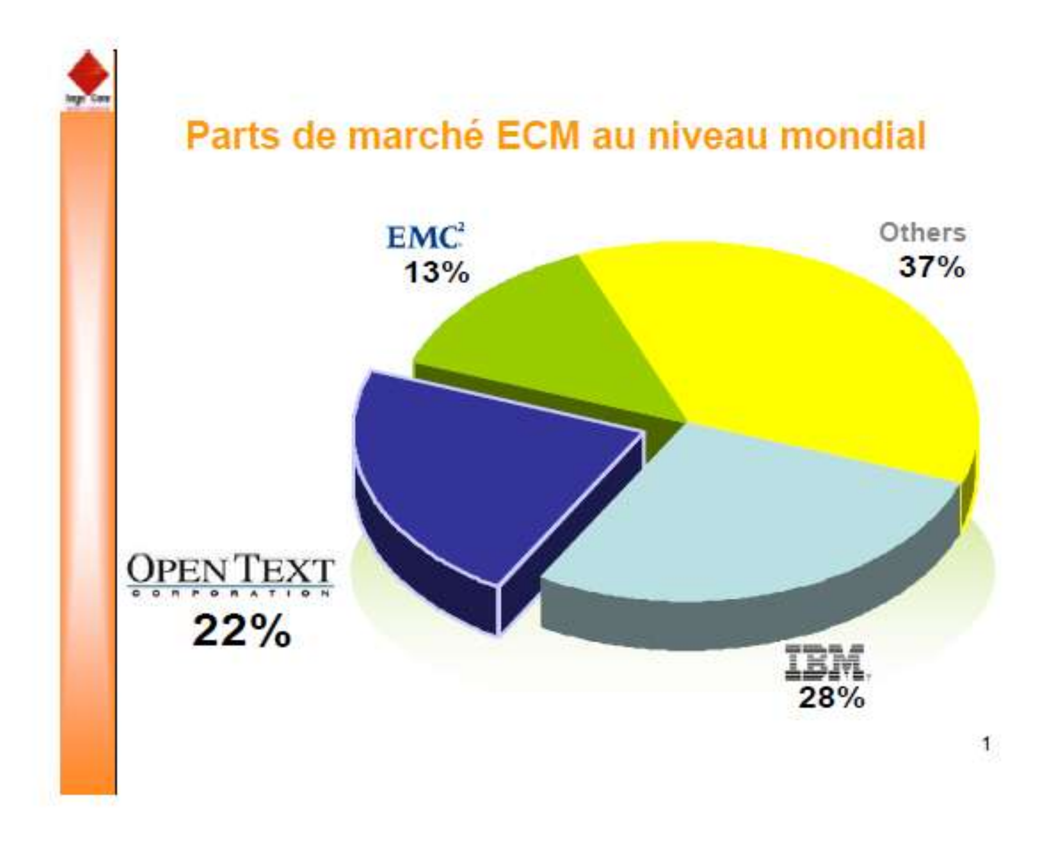


Figure 3.2 : Graphe des éditeurs GED [5]



3.3.2 eDOCS :

eDOCS est une GED offerte par l'éditeur OpenText. Elle gère le contenu de l'entreprise afin d'optimiser l'efficacité opérationnelle, d'éliminer les tâches redondantes et d'accélérer la localisation de la connaissance.

Les principales caractéristiques de la solution eDOCS de gestion de contenu sont les suivantes :

- Référentiel unique : Un seul et même référentiel est utilisé pour le stockage de tous les types de contenu d'entreprise.
- Mise en œuvre des procédures qualité : chaque création de document peut s'effectuer à partir des modèles qui en décrivent la structure et le formalisme.
- Indexation métier : Tous les documents du système se voient attribuer des propriétés métiers propres à l'organisation accueillante.
- Indexation plein texte : de tout le contenu géré par le système.
- Convivialité de l'interface : Les contributeurs ont accès à la base documentaire directement à partir des outils de production en plus de l'interface Web, ce qui permet des gains de productivité conséquents.
- Sécurité renforcée : l'accès aux applications et aux ressources est contrôlé par un modèle de sécurité extrêmement robuste.

Cette solution offre donc à l'entreprise la possibilité de rechercher, classifier, catégoriser et analyser l'information de manière stratégique et en toute sécurité. Elle offre un accès rapide à l'information et permet une exploitation optimale du capital d'information de l'entreprise.

eDOCS est la première GED que nous avons implémenté. Il fallait préparer l'environnement de l'installation et configurer le moteur de base de données pour qu'il héberge la base du logiciel et le serveur IIS pour assurer une connexion via des clients distants. Nous avons créé ensuite les dossiers et les comptes d'utilisateurs du logiciel. Une fois l'installation s'est achevée nous avons livré un guide détaillant les étapes à suivre pour une prochaine installation.





Figure 3.3 Aperçu de la GED eDOCS

Cette figure donne un aperçu de la page d'authentification d' eDOCS, il faut donc entrer le nom d'utilisateur, le mot de passe et le nom du réseau pour accéder à la GED et bénéficier des fonctionnalités qu'elle offre.

3.3.3 CLM

Content Lifecycle Management est la deuxième GED implémentée, produit de l'éditeur OpenText. C'est un système de gestion électronique des documents qui offre une gamme complète de fonctions permettant l'intégration transparente avec d'autres systèmes et applications.

CLM permet l'automatisation et l'application de la logique métier grâce à des Workflows graphiquement construits et directement élaborés. Il permet aussi de contrôler le flux d'informations contenues dans les emails et s'intègre parfaitement à Microsoft Outlook, Groupwise et Lotus Notes, ce qui permet aux utilisateurs de transférer des emails directement de l'application vers le référentiel CLM.

OpenText CLM met à la disposition de ses clients un moteur de recherche pour les aider à trouver et accéder aux informations, il permet de construire des bases de connaissances de



grande taille et interrogeables en indexant tous les objets. La recherche simultanée en texte intégral et sur les catégories et attributs fait également partie des fonctionnalités de CLM.

CLM propose donc un système puissant de gestion documentaire complètement intégré qui permet de gérer l'ensemble des documents y compris les documents critiques de l'entreprise. Il est possible d'accéder, de stocker et de gérer des millions d'informations organisées au sein d'une structure hiérarchique. Le système de contrôle des versions et de suivi des actions garantit une information à jour et précise. Les fonctions puissantes de recherche permettent aux utilisateurs de retrouver facilement l'information. Grâce aux classifications et aux métadonnées, les thèmes et les experts du sujet concerné sont rapidement identifiés.



Figure 3.4 Aperçu de CLM

Cette figure montre la page d'authentification de la GED CLM, il faut entrer les informations nécessaires pour accéder aux fonctionnalités offertes par ce produit. CLM permet d'automatiser les flux de données grâce à l'outil de réalisation des Workflow. Cette fonction distingue CLM de eDOCS, CLM automatise les processus de gestion des documents au sein de Creative alors qu'eDOCS n'offre pas cette fonctionnalité et c'est pourquoi nous avons choisi d'implémenter et de paramétrer CLM. Nous avons fourni des guides d'utilisation et d'installation pour les prochaines implémentations.



Avec CLM nous avons réalisé les différents Workflow des courriers entrants et sortants de Creative dont voici des exemples :

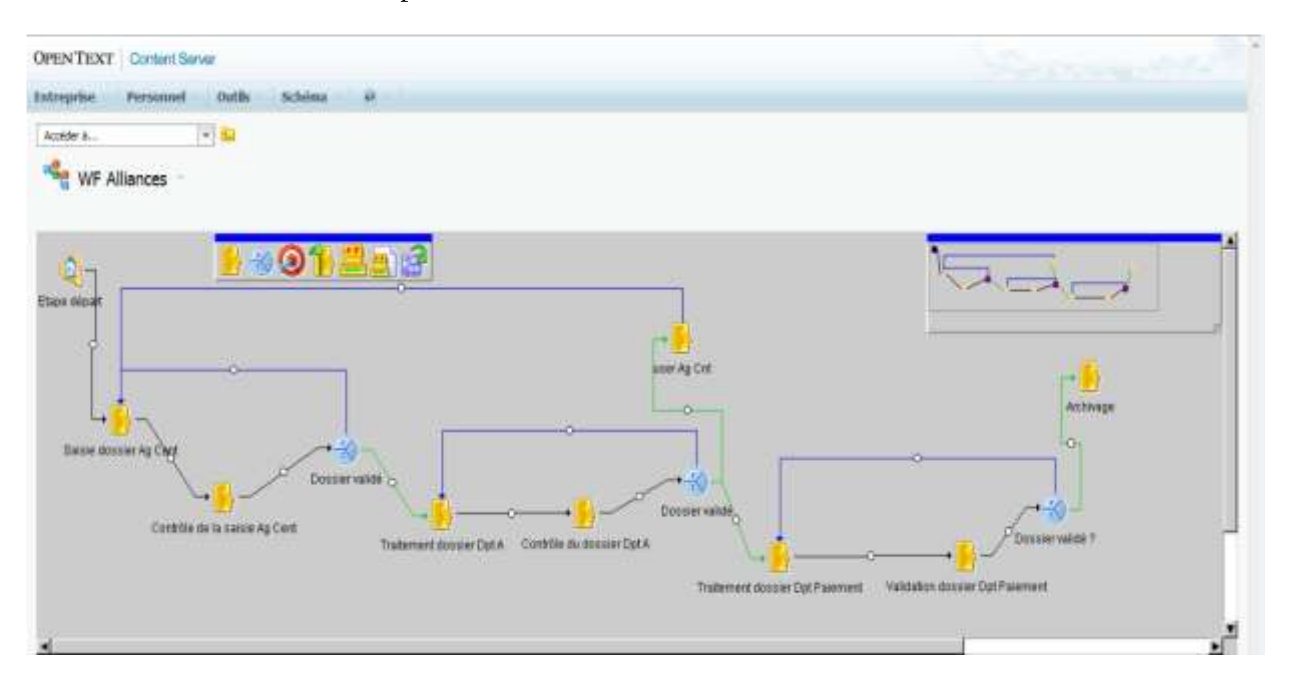


Figure 3.5 Workflow 1

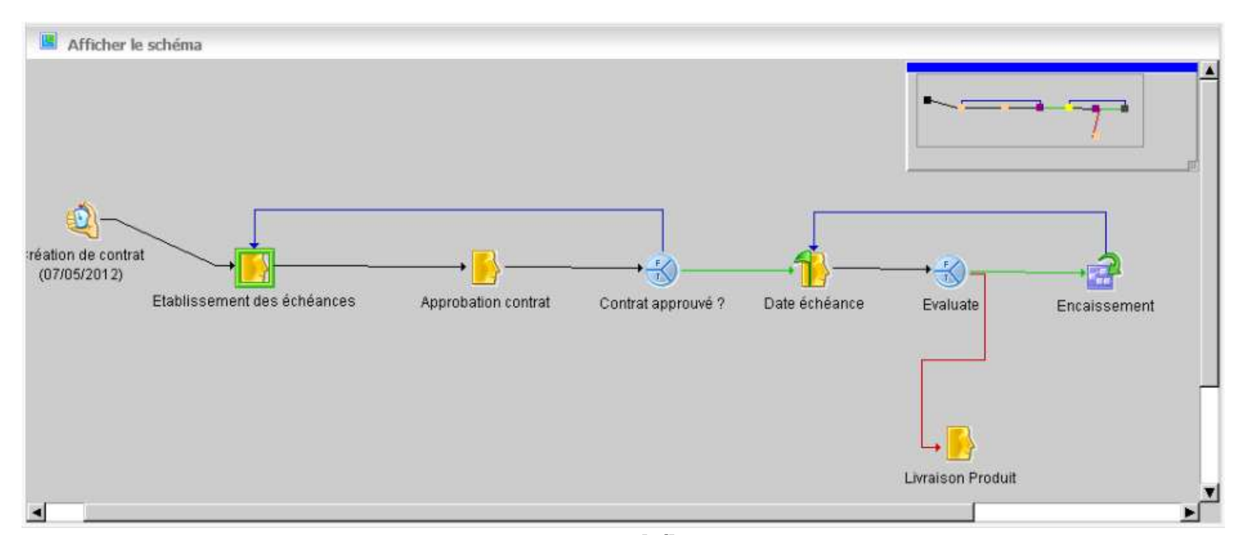


Figure 3.6 Workflow 2

3.4 Conclusion:

Ce chapitre a identifié le système GED à installer et puis à connecter avec d'autres systèmes de l'entreprise, ceci après une étude détaillée des solutions GED les plus connues au marché. Les GED de l'éditeur OpenText sont donc installées et paramétrées tout en livrant des guides d'installation et des manuels d'utilisation. Le chapitre qui suit définie l'architecture sous laquelle le connecteur sera mis en place.



Chapitre 4 : Architecture SOA et Web service

Ce chapitre présente l'architecture de la mise en place de la solution d'intégration des applications qu'est l'architecture orientée service. Ainsi que la première solution d'intégration qui est basée sur les web services.



4.1 Architecture orientée service :

La construction des systèmes d’information s’est le plus souvent réalisée de façon organique, ils sont en plus souvent interconnectés avec des systèmes externes à savoir les systèmes de gestion électronique des documents, ils sont donc adossés à des technologies hétérogènes, rarement interopérables. Rapidement et pour répondre aux besoins croissants d’informatisation des procédures, les problématiques d’intégration de systèmes ont émergé, et avec elles deux questions centrales :

- Comment déclencher, en réponse à un événement dans un sous-système donné, un traitement dans un autre sous-système qui lui est étranger ?
- Comment assurer la consistance et la propagation des données entre plusieurs sous-systèmes ?

SOA n'est pas une technologie, ni une recette, encore moins un produit. C'est une façon de penser et de concevoir le Système d'Informations. A ce titre, les enjeux organisationnels de sa mise en œuvre sont souvent des défis autrement plus difficiles à relever que les enjeux techniques qui la sous-tendent.

Comme son nom le suggère, l'élément clé de SOA est le Service. Un Service est un composant logiciel distribué, exposant les fonctionnalités à forte valeur ajoutée d'un domaine métier. D'un strict point de vue technique, un service possède les caractéristiques suivantes :

- Il propose une interface connue et pérenne.
- Il est logiquement unique (c'est -entre autre- ce qui le distingue d'un composant, qui peut être instancié plusieurs fois).
- Il est invocable à distance (c'est une notion classique des architectures distribuées).
- Il est localisable (à terme, il existe donc un annuaire permettant aux clients de le localiser).

Chaque domaine métier est responsable des services qu'il propose ; il est propriétaire des données et doit se conformer au contrat de service qu'il a publié ; il est en charge de la maintenance et de l'évolution du service. Un service n'a de sens que s'il apporte une valeur métier à l'organisation.

Quelle que soit la méthode choisie, la mise en œuvre d'une SOA nécessite un pilotage transverse, articulé autour des besoins métiers. La construction de l'architecture doit se baser



sur les problématiques métier qu'elle tend à résoudre ; les besoins techniques sont inféodés aux besoins métier.

La formalisation du paradigme SOA a clairement eu le mérite de replacer le métier au centre de l'architecture du SI. Mais SOA reste aussi un projet d'intégration à grande échelle. Comment dès lors éviter les chaussetrappes des grands projets d'architectures distribuées ou intégrées, et en particulier le couplage technique et fonctionnel entre consommateurs et fournisseurs de services ? [11]

Ainsi et afin de répondre à ces problématiques, il est nécessaire de trouver une solution d'intégration sachant que les architectures orientées services sont, dans leur dimension technique, des projets d'intégration à grande échelle. En cela, elles doivent s'appuyer sur une solution d'intégration robuste et souple, permettant d'éviter les écueils du couplage technique et fonctionnel.

4.2 Motivations de la SOA :

Le processus d'un document est déclenché lors de son arrivée au système GED, il parcoure la chaine de traitement au niveau de ce système pour qu'il soit transféré au système cible.

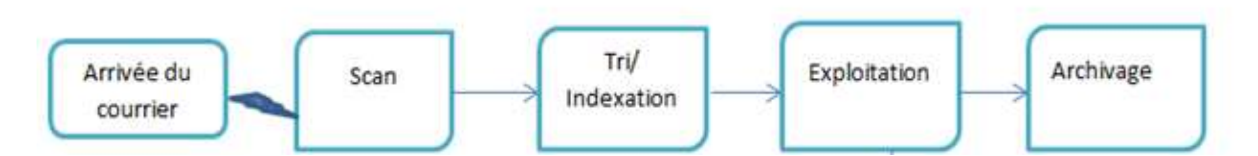


Figure 4.1 : Cycle de vie d'un document au sein de la GED

Une fois le courrier arrive à l'entreprise, il est reçu par le bureau d'ordre, la personne concernée le scanne et l'introduit dans le système de gestion électronique des documents. Dès son injection dans la GED le document subit le processus de traitement électronique, il est référencé et organisé selon les normes par des attributs pour le retrouver plus facilement en lui attribuant une classification ; le critère de recherche peut être l'auteur, le type ou le responsable. Il est ensuite validé pour qu'il soit transmis vers le système destinataire et archivé dans le coffre-fort électronique.



CYCLE DE VIE DES DOCUMENTS

1. Dématérialiser les flux entrants — Applications métiers; Numéri-sation; Bureautique Messagerie; Télé-services

2. Organiser et traiter l'information numérique — Distribution /Traitement; Partage / Recherche; Intégration App. métiers

3. Valider et signer les documents produits — Visas; Signature

4. Transmettre les documents — Transmission sécurisée; Intranet / Internet / Extanet

5. Pérenniser et garantir l'intégrité des documents — Archivage électronique

Figure 4.2 : Cycle de vie des documents dans la GED détaillé. [6]

De ce fait, la partie concernée par cette étude est la quatrième phase du cycle. En effet il s'agit d'automatiser la connexion entre le système GED et un ou plusieurs systèmes d'informations distants. Ces systèmes sont bâtis sur des technologies différentes, le connecteur relie donc la GED avec des applications inconnues pour lui.

L'architecture orientée service offre la possibilité de connecter des systèmes hétérogènes, en se basant sur les services qui servent d'API de l'application et restent ainsi les systèmes à connecter comme étant des boîtes noires.

La première solution d'intégration qui vient à l'esprit en adoptant une architecture orientée service est le web service. En conséquence, la suite de ce chapitre détaille la solution des web service en présentant les protocoles de communication et en choisissant un seul vers la fin du chapitre.



4.3 Les Web Services :

Un Web service est une interface d’échange de données entre deux applications (API) qui s’effectue par le biais d’inter ou intra Net. Il transfère les données vers les applications clientes via des fichiers XML. Le dialogue à distance entre ces applications se fait indépendamment des plateformes et des langages sur lesquelles elles reposent. Pour ce faire, les services web s’appuient sur un ensemble de protocoles internet très répandus (SOAP, REST), afin de communiquer. Cette communication est basée sur le principe de demandes et réponses.

Les services web s’appuient sur un ensemble de protocoles standardisant les modes d’invocation mutuels de composants applicatifs j’en cite SOAP et REST.

4.3.1 SOAP

SOAP (Simple Object Access Protocol) est un protocole qui permet d’échanger des messages XML entre applications. Il définit un ensemble de règles pour structurer des messages qui peuvent être utilisés dans de simples transmissions unidirectionnelles, mais il est particulièrement utile pour exécuter des dialogues requête-réponse. Il n’est pas lié à un protocole de transport particulier mais http est populaire. [7]

a- Fonctionnement

L'approche du développeur SOAP est d'encapsuler la logique de requête de la base de données, destinée à un service, dans une méthode (ou fonction) en C ou VB ou Java etc. et ensuite, de démarrer un processus qui écoute les requêtes adressées à ce service (un process listener), ces requêtes étant formulées dans un format SOAP et contenant le nom du service et les paramètres requis. Comme mentionné, la couche transport peut être HTTP mais pourrait tout aussi bien être SMTP ou autre. Puis, le listener, typiquement écrit dans le même langage que la méthode du service pour plus de simplicité, décode la requête SOAP entrante et la transforme en un appel de la méthode. Il récupère le résultat de l'appel de la méthode, l'encode dans un message SOAP (la réponse) et le renvoie au demandeur. Conceptuellement, cela donne : [12]



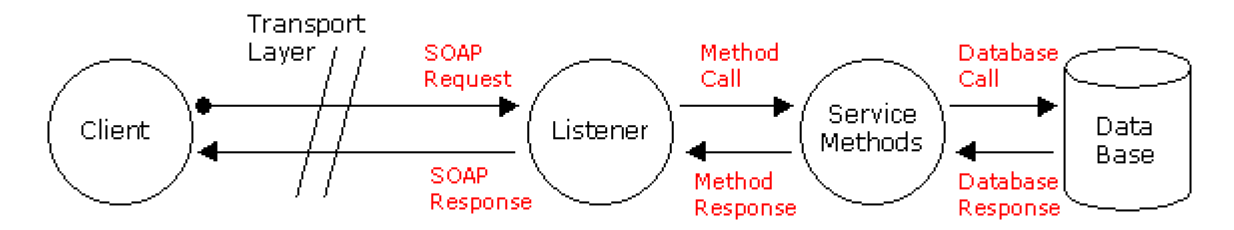


Figure 4.3 : Connecteur SOAP

4.3.2 REST

REST propose une vision orientée ressource des Web services. C'est à dire qu'avec REST les variables ce sont les objets (ou ressources). Selon l'approche REST, chaque ressource est identifiable de manière non ambiguë et unique via son URI (Universal Resource Identifier).

L'interface du service web à réaliser reçoit les métadonnées à transférer au SI, effectue le traitement nécessaire en adoptant le protocole REST et envoie les données au SI. Ainsi les inputs et les outputs du système étudié sont :

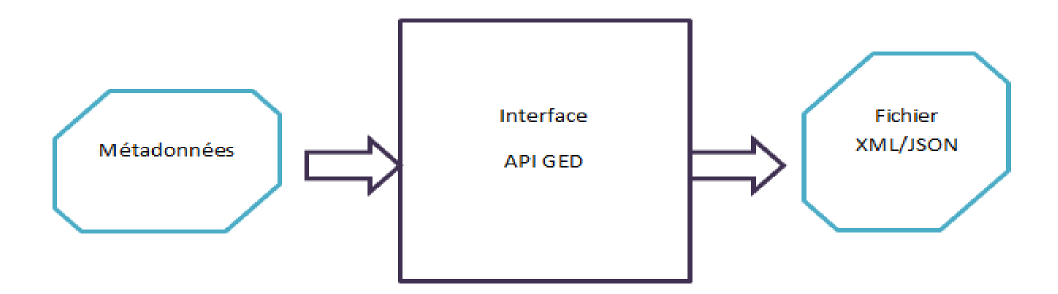


Figure 4.5 : Interface REST

Avec REST pour une ressource donnée, nous proposons un groupe de services (ou opérations) de base. Idéalement toujours les mêmes : créer, consulter, modifier et supprimer (CRUD). Les opérations à réaliser sur ces ressources, coïncident avec les méthodes natives de HTTP :



Correspondance HTTP - REST		
HTTP	**REST**	**Description**
GET	Consulter	Retourne une représentation XML de la ressource. L'état de la ressource est inchangé.
POST	Créer	Créé une nouvelle ressource.
PUT	Modifier	Met à jour une ressource existante.
DELETE	supprimer	Supprime la ressource du système.

Tableau 4.1 : Les méthodes REST

Le service REST doit respecter les « conventions » suivantes :

- Toutes les ressources devant être exposées au travers du service doivent être correctement identifiées, et de manière unique. Chaque ressource devra se voir assigner une URL qui devra être de la forme : http://www.site.com/contenus/XXX avec XXX est un entier identifiant la ressource.
- Les ressources doivent être catégorisées selon leurs possibilités offertes à l'application cliente : ne peut que recevoir une représentation GET ou modifier/créer une ressource POST/PUT, DELETE
- Chaque ressource doit faire un lien vers les ressources liées
- La manière dont fonctionne le service sera décrite au sein d'un document WSDL, ou simplement HTML

4.3.3 Etude comparative SOAP VS REST :

	REST	SOAP
Point commun	-Fonctionne sur http : protocole simple, bien connu, il n'est pas bloqué par les pare-feu. -Message dans le format XML (REST n'est pas limité à XML : utilise aussi JSON : format encore plus léger, moins verbeux)	
Destiné pour	Des opérations simples : CRUD	Des opérations plus complexes
Sécurité	Il n'est pas sécurisé mais il est tout aussi facile de sécuriser un service RESTful (https).	Sécurisé
Administration	Administration web	Administration EAI



Informatique	Tendance léger : REST se met en place comme une page Web, ce qui ouvre la porte à des développements rapides, avec des cycles courts, au prototypage, aux tests possibles avec un simple navigateur et à l'utilisation de langage de scripts	Tendance lourde : SOAP se met en place comme le déploiement d'un objet, ce qui ouvre la porte à une phase préalable de modélisation, des développements aux cycles plus longs et à l'utilisation de langages de programmation à typage statique.

Tableau 4.2 : Etude comparative REST vs SOAP

	REST	SOAP
Avantages	-Leger et simple (message court, facile à décoder) -Auto-descriptif : URL intuitive unique pour chaque ressource. -L'utilisation d'URI comme représentant d'une ressource, permet la mise en place de serveurs cache. -L'absence de gestion d'état du client sur le serveur conduit à une consommation de mémoire inférieure, une plus grande simplicité et donc à une capacité plus grande de répondre à un grand nombre de requêtes simultanées. - ne pas ajouter une couche d'abstraction à des données qui n'en ont pas forcément besoin.	- Extensible -Indépendant du langage -Indépendant de la plate-forme - Supporte le transactionnel -Il est formel : les méthodes à distances et leurs résultats sont bien spécifiées via des contrats -Il est fiable : dispose des mécanismes pour garantir la fiabilité de la transmission et de la réception de données. -standardisé -sécurisé
Inconvénients	-Nécessité pour le client de conserver localement toutes les données nécessaires au bon déroulement d'une requête, ce qui induit une consommation en bande passante réseau plus grande. -Manque de complexité exigée pour les transactions d'affaires : sécurité, support transactionnel - Considéré en quelque sorte comme un artifice du web 2.0 plutôt qu'une vraie solution d'entreprise.	-Complexe et verbeux pour les besoins ordinaires -Il nécessite une compréhension claire des APIs spécifiques au site. Son utilisation demande du temps et de la documentation. -De par le nombre d'informations qu'impose le format XML, SOAP peut alourdir les échanges -Il décrit la manière dont les applications doivent communiquer entre elles, certains considèrent que le couplage reste fort entre le serveur et ses clients.

Tableau 4.3 : Avantages et Inconvénients



Synthèse :

Il est généralement admis que l'objectif de SOAP était seulement d'aller à mi-chemin dans l'utilisation du Web comme un modèle pour la distribution de données : protocole HTTP et format XML. REST a poursuivi le chemin en ajoutant le dernier élément qui manquait à SOAP : le simple et efficace schéma d'adressage URL au sein de l'espace de nommage global du Web. N'importe qui sachant utiliser le Web peut utiliser REST.

Beaucoup de caractéristiques pour lesquelles SOAP est préféré existent en fait dans REST (bien que cela prenne un peu plus d'effort initial pour les mettre en œuvre) : la sécurité via HTTPS, l'authentification via basic Auth, les sessions via les cookies. Cependant, jusqu'à ce que ces besoins et leurs solutions deviennent bien connus et faciles à aborder par les développeurs, SOAP aura sans aucun doute sa place.

4.4 Conclusion :

L'architecture SOA est une réponse très efficace à la problématique d'intégration des applications comme dans le cas de ce projet, elle offre des avantages de réutilisabilité, d'interopérabilité et de réduction de couplage entre les différents systèmes. Les SOA ont été popularisées avec l'apparition du standard Web Service. Les Web services constituent une première solution d'intégration qui répond à la problématique du projet. Et dans le chapitre qui suit nous allons présenter d'autres solutions reposant toujours sur l'architecture SOA.



Chapitre 5 : Les solutions d'intégration GED – SI

Après avoir détaillé la première solution qui est les web services, ce chapitre présente d'autres solutions d'intégration ; des connecteurs permettant de relier le système de gestion électronique des documents avec d'autres systèmes distants basés sur l'architecture orientée service.



L'intégration permet aux applications de gestion de contenu de s'intégrer, de converser avec d'autres applications, le plus souvent productrices ou utilisatrices de contenus.

Le chapitre précédant détaille la première solution d'intégration basée sur les web services, et dans ce qui suit d'autres propositions de solutions :

- ✓ MOM
- ✓ EAI
- ✓ ESB

5.1 Middleware orienté message :

5.1.1 Qu'est-ce qu'un middleware

La définition du Middleware généralement admise est la suivante :

Un Middleware est un logiciel de communication qui permet à plusieurs processus s'exécutant sur une ou plusieurs machines d'interagir à travers un réseau.

Un middleware permet aux applications d'interopérer y compris lorsqu'elles tournent sur des serveurs différents, interconnectés par un réseau. Parmi les middlewares qui permettent l'interopérabilité entre applications homologues (de même nature), on peut distinguer deux grandes familles :

- ➢ Les middlewares qui permettent l'invocation synchrone de fonctions et méthodes.
- ➢ Les middlewares d'échange asynchrones, qui sont principalement à base de messages, ce sont les Middlewares orientés message.

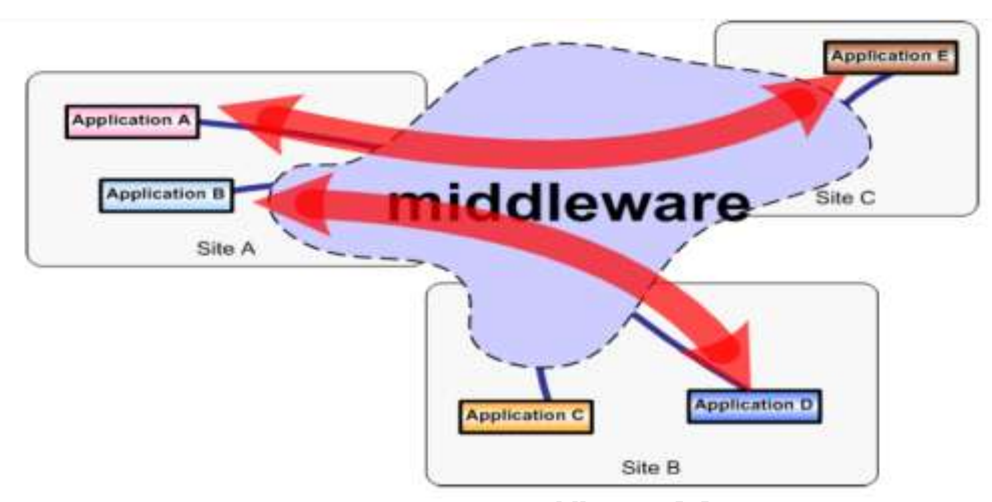


Figure 5.1 : Middleware [8]



5.1.2 Les Middlewares Orientés Messages ou MOM :

Middlewares orientés message, est une architecture basée sur un outil d'échange qui permet à des applications de communiquer en échangeant des messages. Une application « X » adresse un message à une autre application « Y » qui n'est pas nécessairement sur le même serveur. L'application « X » confie son message au MOM, qui se charge de son acheminement vers l'application réceptrice.

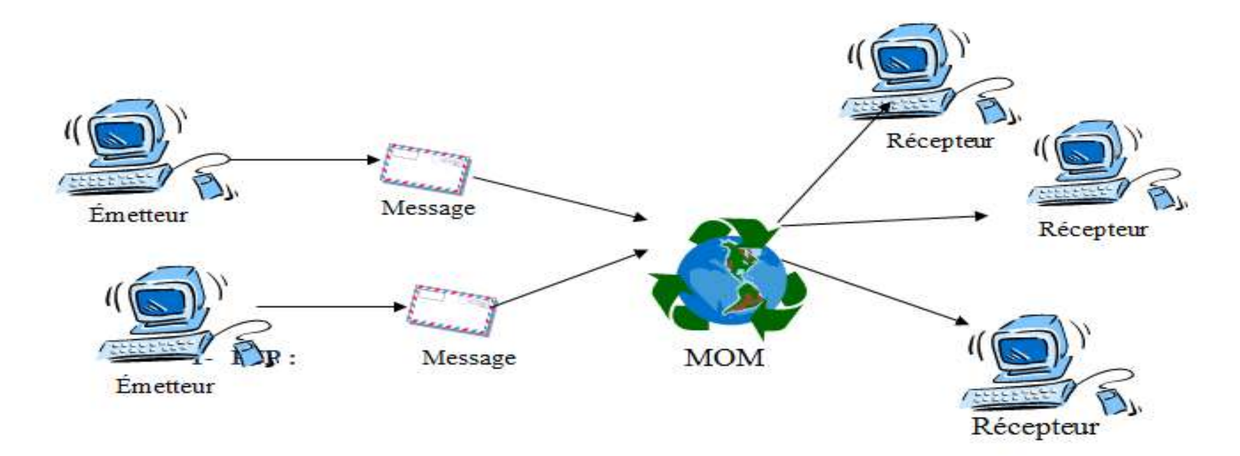


Figure 5.2 : Représentation d'un MOM [8]

L'objet véhiculé par le MOM entre les services est appelé message. Mais rien n'est imposé quant à ce que représente ce message, sa taille, ou encore le format des données qu'il véhicule. Le MOM ne s'intéresse pas au contenu du message, il ne fait que le transmettre, et il le remet au destinataire sans y avoir apporté de changement.

5.1.3 Les brokers

Les Brokers sont des programmes gérant le flux de messages. En d'autres termes, un MOM est composé d'un ou plusieurs brokers. C'est avec les brokers que les applications clientes communiquent, au travers de l'API. Un Broker est un serveur au sens logiciel du terme, c'est-à-dire un processus qui est à l'écoute des requêtes qui peuvent lui être adressées par d'autres processus, les applications clientes. [8]

5.1.4 MOM et SOA

L'architecture des MOM repose sur des APIs du côté du client à travers lesquelles le serveur MOM communique avec ses clients. Ces APIs constituent des services spécifiques à



l'application, les fonctions de l'entreprise sont externalisées comme des services via des interfaces stables définis avec un contrat, ils sont faiblement couplés et interconnectés via des standards comme le MOM dans ce cas.

5.2 Intégration d'applications d'entreprise : EAI

5.2.1 Définition

L'intégration d'applications d'entreprise est une architecture intergicielle permettant à des applications hétérogènes de gérer leurs échanges. On la place dans la catégorie des technologies informatiques d'intégration métier *(Business Integration)* et d'urbanisation. Sa particularité est d'échanger les données en temps réel.[9]

Par extension, l'acronyme EAI désigne un système informatique permettant de réaliser cette architecture en implémentant les flux inter applicatifs du système d'information.

5.2.2 Les fonctionnalités d'un EAI :

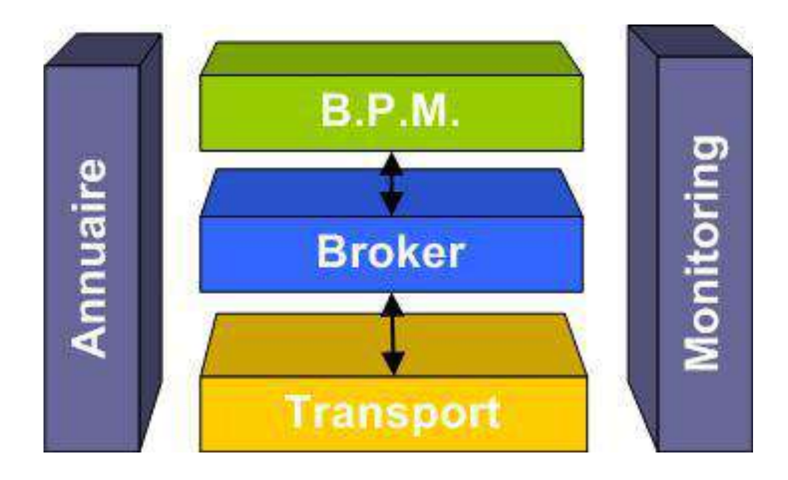


Figure 5.6 Les fonctionnalités d'un EAI [9]

Les fonctionnalités offertes par les EAI sont : le transport des données via les MOM, la connexion entre applications au travers des connecteurs, la transformation des données et l'orchestration métiers. Autres services techniques sont aussi fournis par les EAI comme le monitoring des données.



5.2.3 Notre point de vue:

Malgré leurs évidentes qualités, les solutions d'EAI souffrent de leur caractère très propriétaire :

- Le protocole utilisé pour les échanges et le transport des messages au sein d'un EAI est propriétaire.
- La technologie interne aux EAI est propriétaire. Ainsi, l'accès aux applications se fait par l'intermédiaire de connecteurs encore largement spécifiques à chaque éditeur malgré des tentatives de standardisation comme JCA1 dans le monde Java (ces connecteurs restant souvent très onéreux).
- Les formats et encodages de données utilisés dans les EAI sont propriétaires.

Outre les problèmes de coûts, ce verrouillage est souvent vécu, parfois injustement, comme un risque excessif concernant un composant aussi stratégique.

5.3 Entreprise Service Bus : ESB

5.3.1 Qu'est-ce qu'un ESB :

Les ESB sont les héritiers directs des EAI – il suffit de consulter la liste des principaux éditeurs d'ESB pour s'en convaincre : Bea, Tibco, Oracle, IBM, etc. sont précisément des acteurs de l'EAI.

Reprenant les caractéristiques architecturales des solutions d'EAI, les ESB se concentrent sur les fonctions d'interconnexion et de médiation, et s'appuient pour cela sur un ensemble de standards parmi lesquels :

- ✓ Les Web Services pour gérer les communications synchrones.
- ✓ XML pour définir les formats des messages
- ✓ JMS2 pour adresser la communication asynchrone avec les MOM.
- ✓ JCA pour la connexion aux progiciels et systèmes exotiques (ERP, CRM, Mainframes, etc.)

5.3.2 A quoi sert un ESB ?

D'un strict point de vue technique, le rôle d'un ESB se résume à la connexion et à la médiation entre les services et applications du Système d'Information.

A ce titre, ses principales responsabilités sont les suivantes :

- Réconcilier des mondes hétérogènes, à l'aide de standards d'interopérabilité ou de connecteurs spécialisés – c'est le rôle classique d'un middleware d'intégration.
- Découpler consommateurs et fournisseurs de services : Un consommateur ne connaît que l'ESB et ne connaît ni les formats ni les protocoles d'échange spécifiques utilisés par le fournisseur du service.

- Tracer les messages qui transitent. Devenant une zone de passage incontournable, l'ESB joue un rôle fondamental dans la traçabilité et le monitoring des traitements.[10]

Entreprise
Compta
Données
Paie
Données
Connecteur communiquant avec l'application par WebServices
Connexion vente-stock
Le service de compta et de paie continuent à communiquer directement par leur connecteur historique
Bus E.S.B.
Connecteur relié directement à l'API de l'application.
Connexion vente-clientèle
Connecteur FTP : L'application continue à travailler par échange de fichiers plat. Aucune modification n'a eu lieu pour l'intégration
Vente
Données
Gestion clientèle
Données
Stocks
Données

Figure 5.7 Architecture ESB [10]



5.3.3 Les risques dans la mise en place d'un ESB :

Un ESB occupe un rôle central dans la mise en place de SOA. Néanmoins, positionner un ESB au sein de l'infrastructure du SI n'est pas le seul critère d'aboutissement. La démarche, la méthode et l'organisation mise en place jouent un rôle tout aussi déterminant.

La mise en œuvre d'un ESB dépend du niveau de maturité d'une SOA. L'utilisation d'un ESB n'est pas nécessaire au démarrage d'une SOA. Sa mise en œuvre s'impose après une réflexion plus large au niveau du SI.

Si l'architecture en place est complexe et mal maîtrisée, l'ajout d'un ESB n'apportera pas de réelles solutions. Il risque même de devenir un point critique, mettant en exergue tous les problèmes et incohérences du SI à un seul et même endroit. Ainsi la démarche, la méthode, l'organisation et l'implémentation des briques d'une SOA sont tout aussi importantes.

5.4 Conclusion :

Ce chapitre a mis en exergue trois solutions d'intégration de la GED avec le SI autre que les web services, sachant que ces solutions reposent aussi sur l'architecture orientée service. Les MOM, l'ESB et l'EAI sont des solutions qui contribuent à l'agilité visée lors de la mise en œuvre d'une architecture orientée services (SOA). Dans le chapitre qui suit une étude sera menée dans une perspective comparative afin d'opter la solution à implémenter.



Chapitre 6 : Etude comparative et solution choisie

Ce chapitre mène une étude dans une perspective comparative afin de ressortir la solution à adapter. Après le choix du connecteur intégrant le système GED avec d'autres applications, une présentation détaillée de la solution clôturera le chapitre.



6.1 Etude comparative

6.1.1 Web service VS MOM

Caractéristique	Web service	MOM
Métaphore	Téléphone (sans répondeur)	Courier
Relation temporelle entre client et serveur	Synchrone	Asynchrone
Nature du dialogue	Requête-Réponse	File d'attente
Etat opérationnel du serveur	Obligatoire	Pas nécessaire
Equilibrage de charge	Au moyen d'un moniteur transactionnel	Politique d'extraction des messages
Support des transactions	Dépend du produit	Dépend du produit
Filtrage des messages	Non	Possible
Performances	Plus efficace car pas de sauvegarde	Lent en cas de sécurisation des messages par écriture sur disque

Tableau 6.1 : Etude comparative MOM vs Web Service

La différence majeure entre web service et MOM est que l'un, web service, fonctionne en mode synchrone, tandis que l'autre, MOM, fonctionne en mode asynchrone. Et c'est le critère qui a permis de choisir la technologie à adapter entre le web service et le MOM, sachant que le cahier de charge exige un connecteur assurant une communication asynchrone, la solution du web service est donc éliminée. Il s'agit donc de choisir entre MOM, EAI et ESB.

6.1.2 EAI VS ESB :

En plus de l'inconvénient du coût, ce qui est plus perturbant encore, la multiplication des fonctionnalités des solutions d'EAI a brouillé leur rôle ; en mélangeant les genres – médiation et orchestration -, les EAI sont devenus une brique trop complexe recouvrant trop de responsabilités dans le SI.

EAI et ESB sont très fréquemment opposés par les analystes. Dans les faits, les produits d'EAI ne sont quasiment plus adaptés. Ils se sont transformés en plusieurs produits qui en reprennent les fonctionnalités : d'une part les ESB pour la médiation et d'autre part les solutions de type BPM pour l'orchestration des processus.



Ainsi le choix entre ces deux technologies s'avère dorénavant facile à effectuer, l'EAI est éliminé à cause des contraintes qu'il engendre (coût d'investissement) et est remplacé par l'ESB qui reprend la plupart de ces fonctionnalités avec des améliorations, la plus frappante est la gratuité, en effet il existe nombreux ESB open source qui font l'affaire.

6.1.3 MOM VS ESB :

L'ESB n'apparaît comme une solution d'intégration pertinente qu'au sein d'une infrastructure complexe d'échange de données. Pour le cas de ce projet, le but est de connecter des SI avec une ou plusieurs GED qui en résulte un système compliqué mais pas complexe, compliqué par ce qu'il englobe plusieurs composants et non pas complexe par ce qu'il ne demande pas une orchestration ou analyse des données à faire circuler entre ces systèmes. En conséquence, la solution la plus adaptée aux critères fonctionnels du projet est le middleware orienté message qui répond parfaitement aux besoins sans excès ni manque de fonctionnalités.

6.2 Les MOMs Open source :

Il existe plusieurs solutions MOM Open Source, les quatre outils les plus crédibles, les plus solides, et les plus pérennes, ceux sur lesquels on peut envisager sans risque de construire une architecture critique pour l'entreprise sont :

- ✓ Active MQ
- ✓ JBoss Messaging
- ✓ Joram
- ✓ Open Message Queue

Le tableau comparatif ci-dessous, se base sur les critères du langage supporté, des protocoles utilisés pour acheminer les données, de la sécurité des transferts et d'autres critères permettant d'évaluer le serveur à choisir. La logique de comparaison consiste à donner '×' si le serveur n'offre pas la fonctionnalité ou marquer par '✓' s'il la supporte mais avec des contraintes ou par '✓✓' s'il la supporte mais sans droit d'administration ou par '✓✓✓' si cette fonctionnalité est supporté avec les droits de lecture écriture et administration.



Critères		Joram	AMQ	OMQ	JBM
Langage	Java	✓✓✓	✓✓✓	✓✓✓	✓✓✓
	C/C++	✓✓	✓✓✓	✓	✗
	JavaScript	✗	✓✓✓	✗	✗
	.Net	✗	✓✓✓	✗	✗
	PHP	✗	✓✓✓	✗	✗
Protocoles	Interne	✓	✓✓✓	✓	✓
	AMQ.P	✗	✓✓	✗	✗
	Open Wire	✗	✓✓✓	✗	✗
	STOMP	✗	✓✓✓	✗	✗
Passerelles fournies	SOAP	✓✓	✓✓	✓✓	✗
	RestFul	✗	✓✓	✗	✗
	Mail	✓✓	✓✓	✗	✗
	FTP	✓✓	✓✓	✗	✗
Interfaces	JavaScript/Ajax	✗	✓✓	✗	✗
	JMS 1.0.2b	✓✓	✗	✗	✓✓
	JMS 1.1	✓✓✓	✓✓✓	✓✓✓	✓✓✓
Intégration	E.J.B	✓✓✓	✓✓✓	✗	✓✓✓
	Spring	✓✓✓	✓✓✓	✗	✗
	Standalone	✓✓	✓✓✓	✓✓	✓
Topologie	Configuration	✓✓	✓✓	✓✓✓	✓✓
	Multi-site	✓✓✓	✓✓	✓✓	✓✓
	Réplication	✓✓	✓✓	✗	✓✓
Gestion des messages	Hiérarchie de Topic	✓✓	✓	✗	✗
	Priorité	✓✓	✗	✗	✗
	Groupe de messages	✗	✓✓✓	✗	✗
	Destination virtuelle	✗	✓✓✓	✗	✗
Administration/ Monitoring	Par messages	✗	✓✓	✓✓	✗
	Interfaces graphique fournie	✓	✓✓✓	✓✓✓	✗
	Interface script fournie	✗	✗	✓✓✓	✗
Sécurité	SSL/ TLS	✓✓✓	✓✓✓	✓✓✓	✓✓✓
	Gestion utilisateur	✓✓✓	✓✓✓	✓✓	✓✓
	Gestion de groupe	✗	✓✓✓	✓✓✓	✓✓✓
	Gestion des droits par domaine	✓✓✓	✓✓✓	✓✓✓	✓✓✓

Tableau 6.2 : Etude comparative Serveur MOM



Nous constatons que le serveur le plus complet est ActiveMQ, c'est donc le serveur choisi pour implémenter la solution des middlewares orientés message.
Sorti en 2004, Active MQ est le MOM open source de la fondation Apache. Il est distribué sous licence Apache 2.0. Il a la possibilité de traiter le message et de le router selon son contenu vers le/les destinataire(s). Il est aussi caractérisé par sa gestion des transactions. La journalisation du « Message Store » permet la reprise sur incident sans perte de données lors d'un « rollback ».

6.3 Implémentation MOM

6.3.1Principe

Les MOM sont basés sur des systèmes de queues de messages persistantes qui permettent de transmettre de manière asynchrone des messages entre des composants qui peuvent ne pas fonctionner aux mêmes instants. Les composants sont donc asynchrones et le message est stocké dans la queue quand le composant destinataire n'est pas accessible.
Le client producteur envoie un message et le client destinataire doit demander sa réception. Le transfert du message et sa persistance sont assurés par le broker.

Les échanges de message sont :

- asynchrones :
- fiables : les messages ne sont délivrés qu'une et une seule fois

6.3.2 JMS

JMS est l'API de Java assurant une communication asynchrone via Message. C'est l'API qui permet à une application d'invoquer les services d'un MOM.
La spécification JMS introduit deux modes de communication :

- ✓ Le mode point à point ou « queue » : Une application envoie des messages à une queue. Une seule des applications connectées reçoit le message. Il peut y avoir plusieurs applications en lecture sur la queue, mais une seule d'entre elles recevra le message.[8]



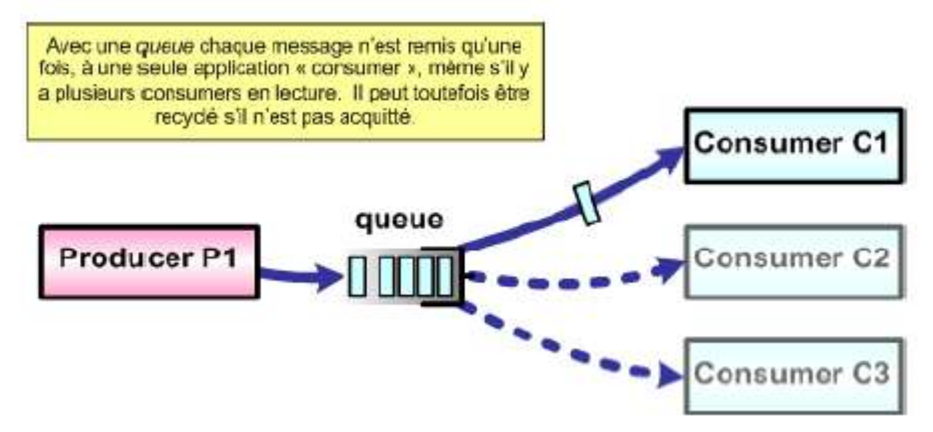


Figure 6.1 Représentation d'une queue MOM [8]

- ✓ Le mode « publish-subscribe » ou « topic » : Dans ce mode les applications s'abonnent à un topic, afin de recevoir les messages. Plusieurs applications peuvent être abonnées à un même topic, et chacune d'elles reçoit une copie des messages.
  C'est donc un échange de 1 vers N, mais qui peut être aussi bien « de P vers N », car plusieurs applications peuvent écrire dans le topic.[8]

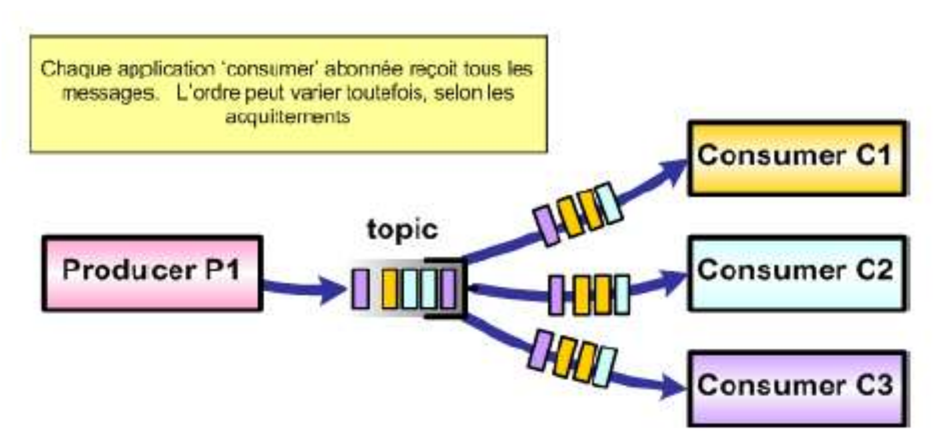


Figure 6.2 Représentation d'un Topic MOM [8]

Le producteur dans notre cas est un client non JMS, en effet la GED est réalisée en .Net et elle communiquera des données à des serveurs distants via le JMS Provider. Le mode point à point ne répond pas à notre besoin direct, alors que le mode Topic est le mieux parlant dans le cas de l'intégration de P applications avec N autres applications.



6.3.3 Mise en œuvre :

Lancer le serveur en tapant la commande : bin\activemq dans le DOS

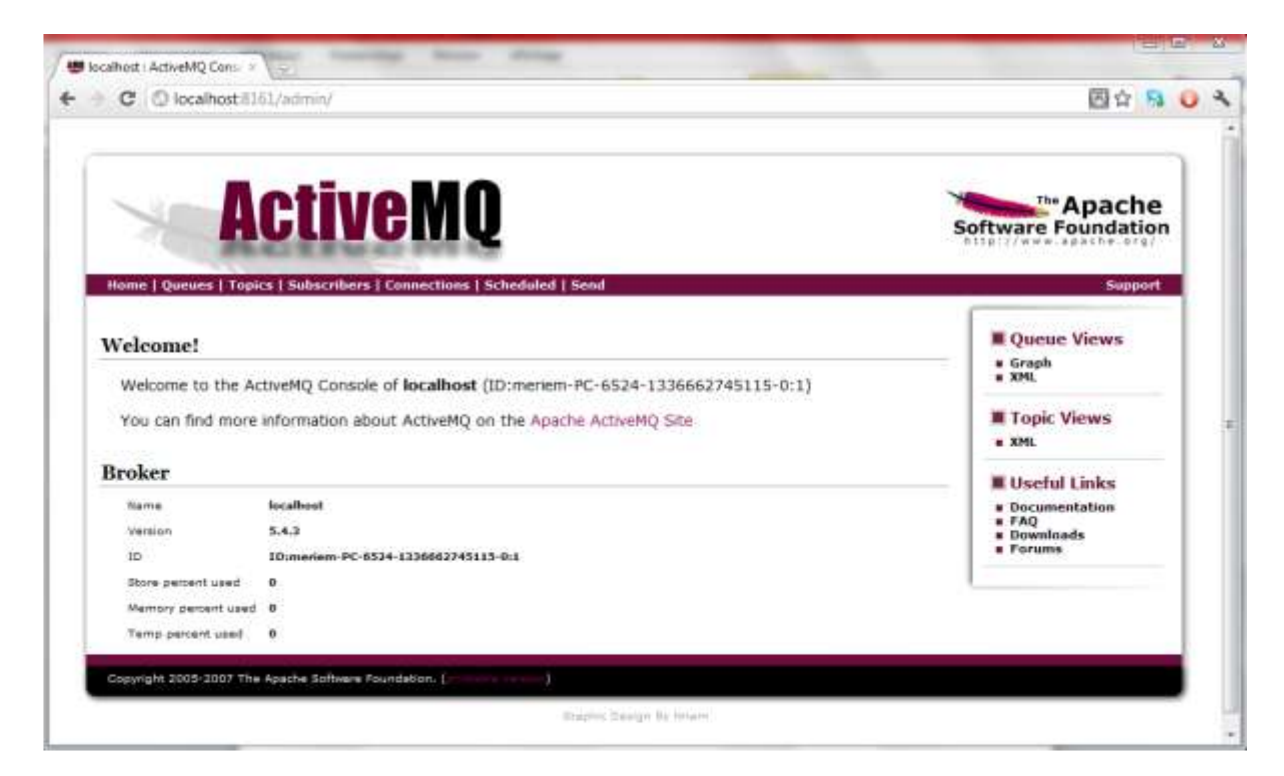
```
Command Prompt
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\meriem>cd /

C:\>cd Apache\apache-activemq-5.4.3

C:\Apache\apache-activemq-5.4.3>bin\activemq
```

Une fois le serveur est lancé, accéder au lien localhost :8161/admin/



C'est la version Web du serveur ActiveMQ, cette interface d'administration de serveur permet de consulter les files d'attente et les Topics du serveur et visualiser leur état. Elle donne aussi la possibilité d'envoyer des messages pour tester le fonctionnement du serveur.

Nous allons tester la faisabilité de la solution MOM, en envoyant un message via un client vers le serveur, et en lançant un autre client qui va demander la réception du message.



Java Monitoring & Management Console - service:jmx:rmi:///jndi/rmi://localhost:1099/jmxrmi

Connection Window Help

Overview | Memory | Threads | Classes | VM Summary | MBeans

JMImplementation
com.sun.management
java.lang
java.util.logging
org.apache.activemq
localhost
Broker
Connector
Queue
Subscription
Topic
ActiveMQ.Advisory.Connection
ActiveMQ.Advisory.Producer.Topic.TopicTe
ActiveMQ.Advisory.Queue
ActiveMQ.Advisory.Topic
TopicTest
Attributes
Operations
browse
browse
browseAsTable
browseAsTable
resetStatistics
sendTextMessage
sendTextMessage
sendTextMessage
sendTextMessage
browseMessages
browseMessages

Operation invocation

java.lang.String sendTextMessage (body Tester le MOM)

MBeanOperationInfo

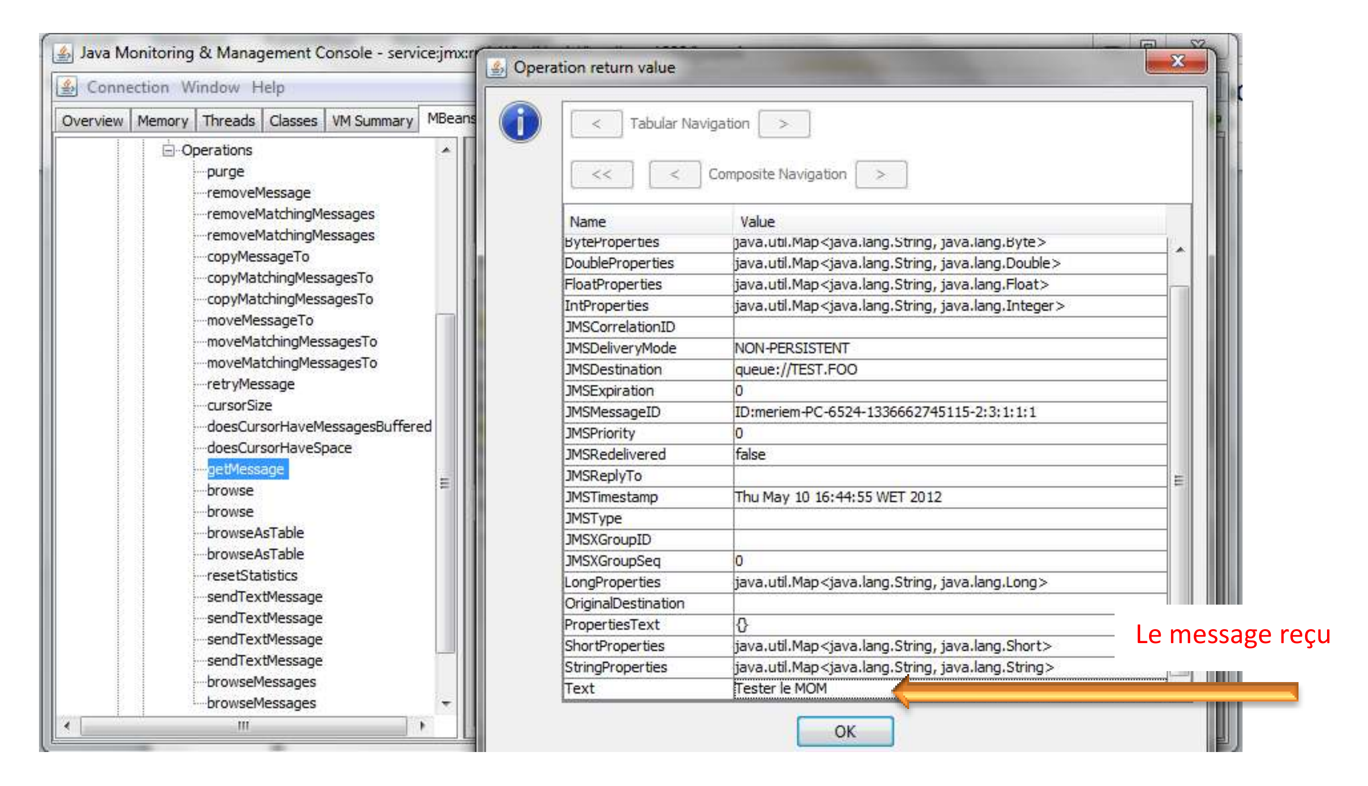
Name	Value
Operation:	
Name	sendTextMessage
Description	Sends a TextMessage to the destination.
Impact	UNKNOWN
ReturnType	java.lang.String
Parameter-0:	
Name	body
Description	
Type	java.lang.String

Descriptor

Name	Value

Cette interface représente le client java avec lequel nous allons tester le fonctionnement de la solution MOM, nous allons envoyer un message de test 'Tester le MOM' par la méthode 'sendTextMessage' du client java Jconsole à destination du serveur MOM lancé auparavant.





Nous avons lancé un deuxième client Jconsole qui n'a pas été active pendant l'envoi du message, et qui demande au serveur MOM de lui transférer des messages s'il les a reçus, cette demande est déclenché par la méthode 'getMessage'. Le deuxième client reçoit alors le message 'Tester le MOM'.

6.4 Conclusion :

Ce chapitre a permis de choisir la solution d'intégration la plus adéquate aux besoins fonctionnels de l'entreprise, répondant aux objectifs et enjeux du projet. Les Middleware orientés messages sont des outils permettant aux applications d'interopérer en échangeant des messages, de manière asynchrone et fiable, et en respectant l'architecture SOA. Ce sont des outils précieux pour construire des plateformes assemblant des applications et composants divers, comme pour gérer l'interopérabilité.

Moins ambitieux que l'ESB, le MOM est un outil relativement simple, extrêmement performant et néanmoins présentant l'extensibilité et la fiabilité requises pour se déployer à l'échelle de l'entreprise.



Conclusion générale:

Au terme de ce travail, je rappelle que mon projet de fin d'études s'est déroulé au sein de la société Creative Technologies. Lors de ce stage de 16 semaines, j'ai pu mettre en pratique mes connaissances théoriques acquises durant ma formation, de plus, je me suis confrontée aux difficultés réelles du monde du travail et du travail en équipe.

Sur le plan professionnel, ce projet était très enrichissant, car il offre une vision globale sur les solutions d'intégration des systèmes d'information avec les systèmes de gestion électronique des documents. Ces systèmes qui constituent un volet informatique d'actualités et d'avenir. Le projet qui vise la mise en place d'un connecteur GED-SI, il s'est donné deux objectifs: premièrement, installer le système GED choisi, le paramétrer et automatiser les cycles des courriers de l'organisme accueillant en réalisant des Workflows. Deuxièmement, étudier les solutions d'intégration du système GED avec le/les SI, et choisir ensuite la solution la plus adaptée aux besoins fonctionnels de l'entreprise répondant ainsi aux objectifs du projet.

Outre le plan professionnel ce projet était d'un bénéfice inestimable en terme de gestion de projet, de connaissances techniques et aussi par rapport au contact avec l'ensemble de l'équipe architecture recherche et développement pour comprendre le besoin et apporter la solution. Un travail au sein d'une entité telle que la CDG, ne peut être que très enrichissant au niveau du savoir et du savoir-faire.

Je pense que cette expérience en entreprise m'a offert une bonne préparation à mon insertion professionnelle car elle fut pour moi une expérience enrichissante et complète qui conforte mon désir d'exercer mon futur métier d'ingénieur dans le domaine des systèmes d'information.



Bibliographie & webographie :

[1] Le livre « Scrum et XP depuis les tranchées » 2008 Henrik Kniberg

[2] http://www.mcnext.com/pole_dotnet/projet-agile/Pages/Scrum.aspx / site sur la démarche d'un projet agile visité en Fevrier 2012

[3] http://www.awt.be/web/res/index.aspx?page=res,fr,fic,100,003 / site sur l'intégration des applications visité en Mars 2012

[4] Livre blanc : « Tout processus métier a sa dématérialisation » : 2010 Christophe Lucot

[5] OpenText : Dossier Fonctionnel « Livelink ECM eDOCS suite 6.0 » 2009

[6] Dématérialisation et gestion électronique de documents : Article sur le journal du net visité en Fevrier 2012

[7] http://www.w3.org/2002/07/soap-translation/soap12-part0.html / site décrivant le protocole SOAP visité en Mars 2012

[8] Livre blanc : Smile Open Source Solution France : http://www.smile.fr/Livres-blancs/Systeme-et-infrastructure/Middleware-orientes-messages , livre qui détaille l'architecture des MOM visité en Mars 2012

[9] http://fr.wikipedia.org/wiki/Int%C3%A9gration_d%27applications_d%27entreprise visité en Avril 2012

[10] http://fr.wikipedia.org/wiki/Enterprise_Service_Bus visité en Avril 2012

[11] Livre blanc : « Comprendre et savoir utiliser un ESB dans une SOA » 2007 Xebia consulté en Mai 2012

[12] http://www.soapuser.com/fr/basics1.html site destiné pour les utilisateurs du protocole SOAP visité en Mars 2012



Annexe : Environnement de développement d'une architecture MOM

1. Concepts

1.1 Client JMS :

Le fonctionnement d'un client typique JMS passe par une phase d'initialisation qui s'enchaine comme suit :

- ❖ Trouver l'objet « ConnectionFactory » par JNDI
- ❖ Trouver un (ou plusieurs) objet Destination par JNDI
- ❖ Créer une Connexion JMS
- ❖ Créer une (ou plusieurs) session avec le Connexion JMS
- ❖ Créer le(s) MessageProducer ou/et MessageConsumer avec la Session et la (les) Destination
- ❖ Demander à la Connexion de démarrer la livraison des messages

Ensuite le client JMS passe par une phase de consommation/production dans laquelle il crée les messages et les envoie ou reçoit les messages et les traite.

1.2 Client NMS :

L'aptitude à faire communiquer des applications hétérogènes fait partie des missions naturelles d'un middleware. Le client NMS est l'implémentation d'un client MOM avec .net.

Active MQ se caractérise par la diversité des langages et environnements supportés, il offre des bibliothèques implémentant les APIs JMS sous différents langages. Pour notre cas Active MQ contient une bibliothèque .Net proposant des interfaces similaires à JMS qui est NMS.

L'API NMS permet de réaliser des applications .Net avec C# , VB ou n'importe quel autre langage .Net en suivant le même style d'API JMS.

1.3 JMS provider

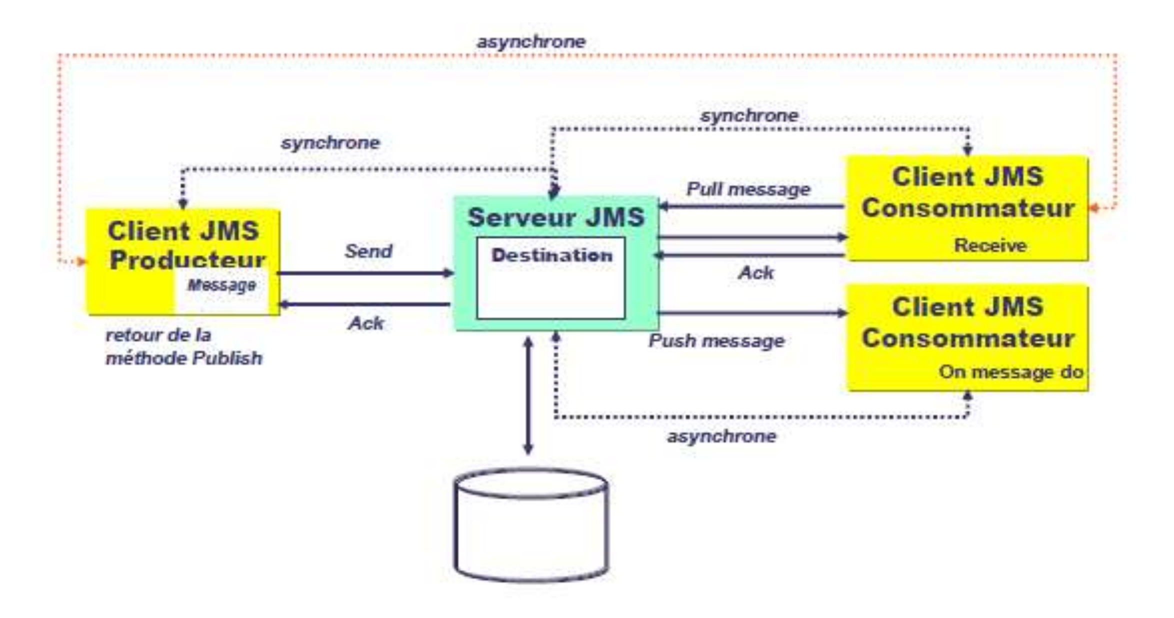
Le broker ou le provider est le serveur chargé de recevoir les messages de chez le producteur et les transmettre vers le consommateur. Il existe deux modes d'envoi de message entre le serveur JMS et le consommateur :

- Mode Push : La consommation est implicite càd qu'une méthode prédéfinie est attachée à la production d'un message (événement) et donc l'occurrence d'un



événement qui est la réception d'un message par le provider entraîne l'exécution de la réaction associée. C'est un modèle Evénement/ Réaction

- Mode Pull : La consommation est explicite càd que les consommateurs programment explicitement l'accès aux messages, en cas d'absence de message le consommateur est mis en attente.



2. Implémentation :

2.1 JMS Provider :

Le JMS provider utilisé est ActiveMQ, il s'agit donc de le télécharger et le configurer pour qu'il joue le rôle du serveur de message.

Etape 1 : Télécharger le zip de ActiveMQ et l'extraire dans un dossier.

Etape 2 : Ouvrir une console de commande DOS, et se situer dans le dossier de activemq

Etape 3 : lancer la commande bin\activemq



```
C:\Windows\system32\cmd.exe - bin\activemq
C:\>cd Apache

C:\Apache>cd apache-activemq-5.4.3

C:\Apache\apache-activemq-5.4.3>bin\activemq
Java Runtime: Sun Microsystems Inc. 1.6.0_23 C:\Program Files (x86)\Java\jdk1.6.
0_23\jre
  Heap sizes: current=15872k  free=14592k  max=506816k
    JVM args: -Dcom.sun.management.jmxremote -Xmx512M -Dorg.apache.activemq.UseD
edicatedTaskRunner=true -Djava.util.logging.config.file=logging.properties -Dact
ivemq.classpath=C:\Apache\apache-activemq-5.4.3\bin\../conf;C:\Apache\apache-act
ivemq-5.4.3\bin\../conf; -Dactivemq.home=C:\Apache\apache-activemq-5.4.3\bin\..
-Dactivemq.base=C:\Apache\apache-activemq-5.4.3\bin\..
ACTIVEMQ_HOME: C:\Apache\apache-activemq-5.4.3\bin\..
ACTIVEMQ_BASE: C:\Apache\apache-activemq-5.4.3\bin\..
Loading message broker from: xbean:activemq.xml
 INFO | Refreshing org.apache.activemq.xbean.XBeanBrokerFactory$1@1ccce3c: start
up date [Sun Apr 08 12:43:30 WET 2012]; root of context hierarchy
 WARN | destroyApplicationContextOnStop parameter is deprecated, please use shut
down hooks instead
 INFO | PListStore:C:\Apache\apache-activemq-5.4.3\bin\..\data\localhost\tmp_sto
rage started
 INFO | Using Persistence Adapter: KahaDBPersistenceAdapter[C:\Apache\apache-act
ivemq-5.4.3\bin\..\data\kahadb]
 INFO | KahaDB is version 3
 INFO | Recovering from the journal ...
 INFO | Recovery replayed 1 operations from the journal in 0.675 seconds.
 INFO | ActiveMQ 5.4.3 JMS Message Broker (localhost) is starting
 INFO | For help or more information please see: http://activemq.apache.org/
 INFO | Listening for connections at: tcp://meriem-PC:61616
 INFO | Connector openwire Started
 INFO | ActiveMQ JMS Message Broker (localhost, ID:meriem-PC-1295-1333889027044-
0:1) started
 INFO | jetty-7.1.6.v20100715
 INFO | ActiveMQ WebConsole initialized.
 INFO | Initializing Spring FrameworkServlet 'dispatcher'
 INFO | ActiveMQ Console at http://0.0.0.0:8161/admin
 INFO | ActiveMQ Web Demos at http://0.0.0.0:8161/demo
 INFO | RESTful file access application at http://0.0.0.0:8161/fileserver
 INFO | Started SelectChannelConnector@0.0.0.0:8161
```

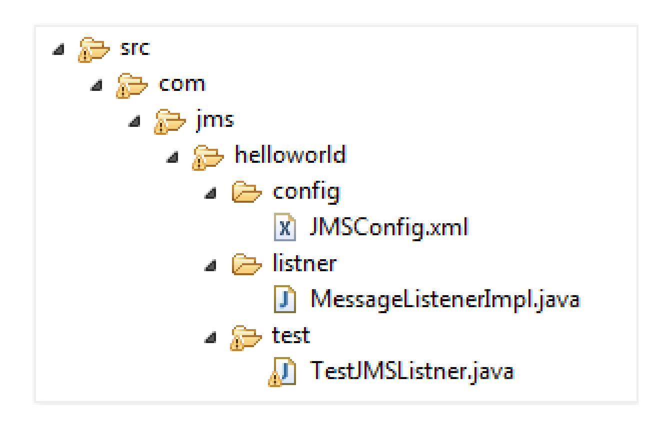
2.2 Client JMS :

Après avoir installé l'environnement java à savoir Eclipse et Spring, il s'agit d'intégrer les lib de Activemq dans le projet.

```
lib
  activemq-all-5.5.0.jar
  activemq-pool-5.5.0.jar
  commons-codec.jar
  commons-httpclient.jar
  commons-logging.jar
  commons-pool-1.5.4.jar
  jstl-1.2.jar
  log4j-1.2.15.jar
  org.springframework.aop-3.1.0.M2.jar
  org.springframework.asm-3.1.0.M2.jar
  org.springframework.aspects-3.1.0.M2.jar
  org.springframework.beans-3.1.0.M2.jar
  org.springframework.build.aws.ivy.jar
  org.springframework.context-3.1.0.M2.jar
  org.springframework.context.support-3.1.0.M2.jar
  org.springframework.core-3.1.0.M2.jar
  org.springframework.expression-3.1.0.M2.jar
  org.springframework.instrument-3.1.0.M2.jar
  org.springframework.instrument.tomcat-3.1.0.M2.jar
  org.springframework.jdbc-3.1.0.M2.jar
  org.springframework.jms-3.1.0.M2.jar
  org.springframework.orm-3.1.0.M2.jar
  org.springframework.oxm-3.1.0.M2.jar
  org.springframework.spring-library-3.1.0.M2.libd
  org.springframework.test-3.1.0.M2.jar
  org.springframework.transaction-3.1.0.M2.jar
  org.springframework.web-3.1.0.M2.jar
  org.springframework.web.portlet-3.1.0.M2.jar
  org.springframework.web.servlet-3.1.0.M2.jar
  org.springframework.web.struts-3.1.0.M2.jar
  slf4j-api-1.5.11.jar
  slf4j-log4j12-1.5.11.jar
  xbean-server-3.3.jar
```

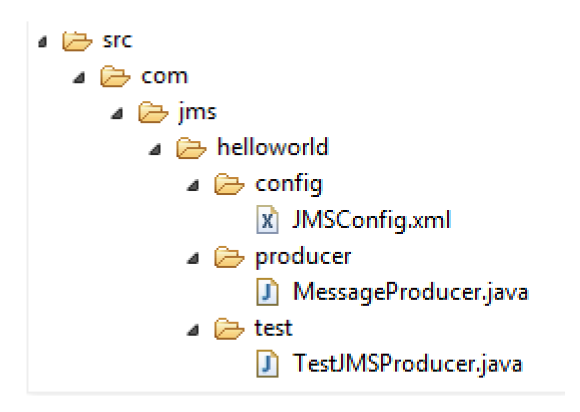


Ensuite de créer l'arborescence suivante :



Le Listener est le consommateur du message.

Le client producteur a l'arborescence suivante :



Après avoir lancé le provider ActiveMQ , démarrez le Listener par compilation de la classe TestJMSListener.java, puis démarrez le producteur TestJMSProducer.java.

2.3 Client NMS :

Après avoir installé Spring .Net qui offre des librairies NMS, il s'agit de créer les classes Listener et Producer basés sur un « Topic ». Dans Visual Studio (2008 par exemple) créer un nouveau projet ayant le nom Core par exemple, ajouter les références suivantes :

- ActiveMQ
- NMS

Créer une classe pour s'abonner et consommer les messages de manière asynchrone. Tout le travail doit être effectué dans le constructeur. L'attribution de l'identifiant au client est nécessaire pour que la souscription soit durable. Il est intéressant de noter qu'il y a un tas de chose qui devra apparaître : la connexion, la session et le message.

Déclarer ensuite un abonné, créer un autre projet avec les références suivantes :

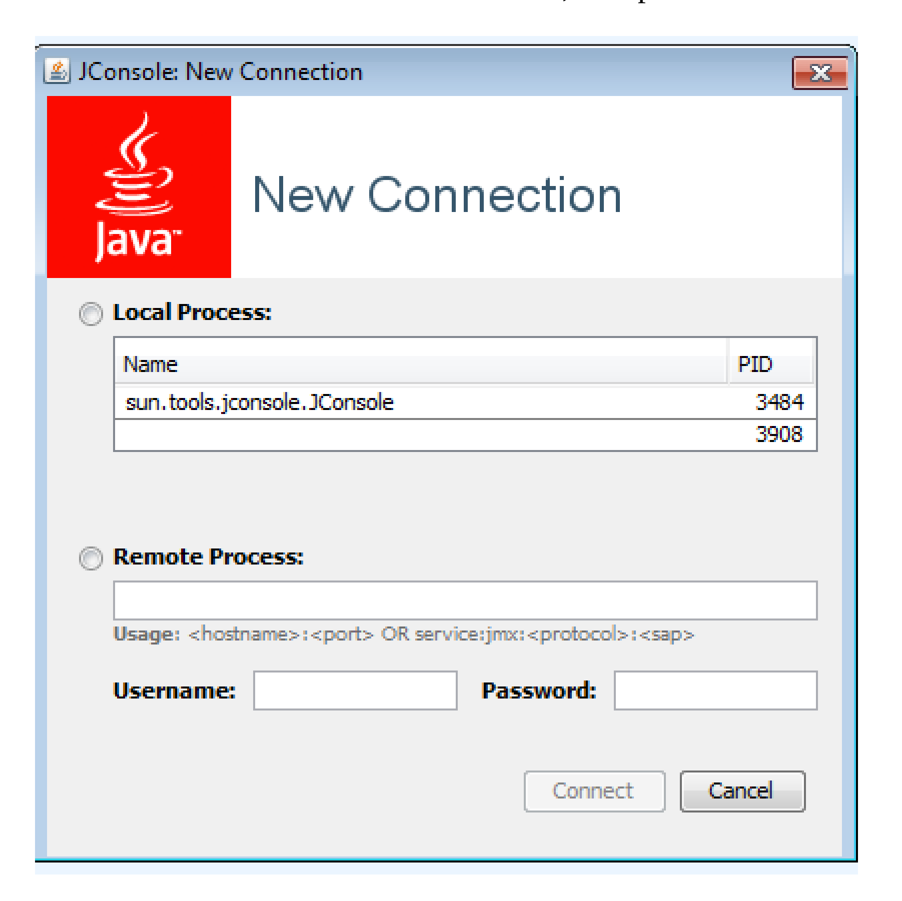
- Spring.Core
- ActiveMQ



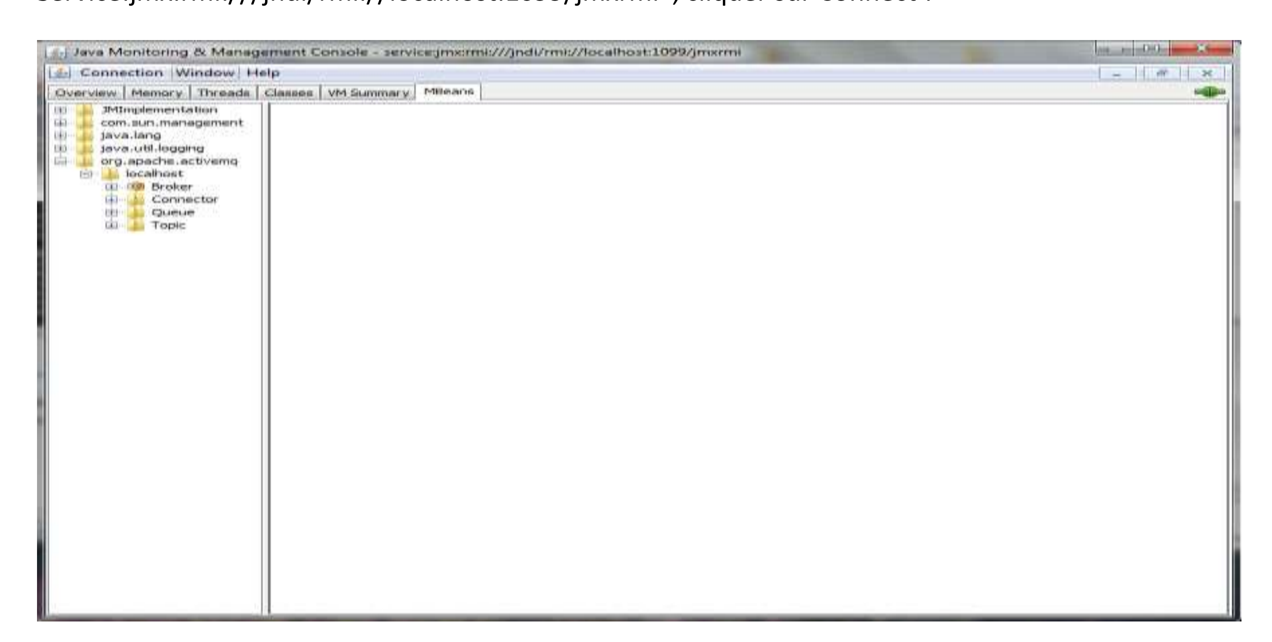
- NMS

Ces dlls peuvent être récupérés du répertoire bin\net\2.0\debug de Spring.Messaging.NMS.

Aller à Home\JDK\bin et lancer Jconsole, cela permet de lancer le Management Console Java

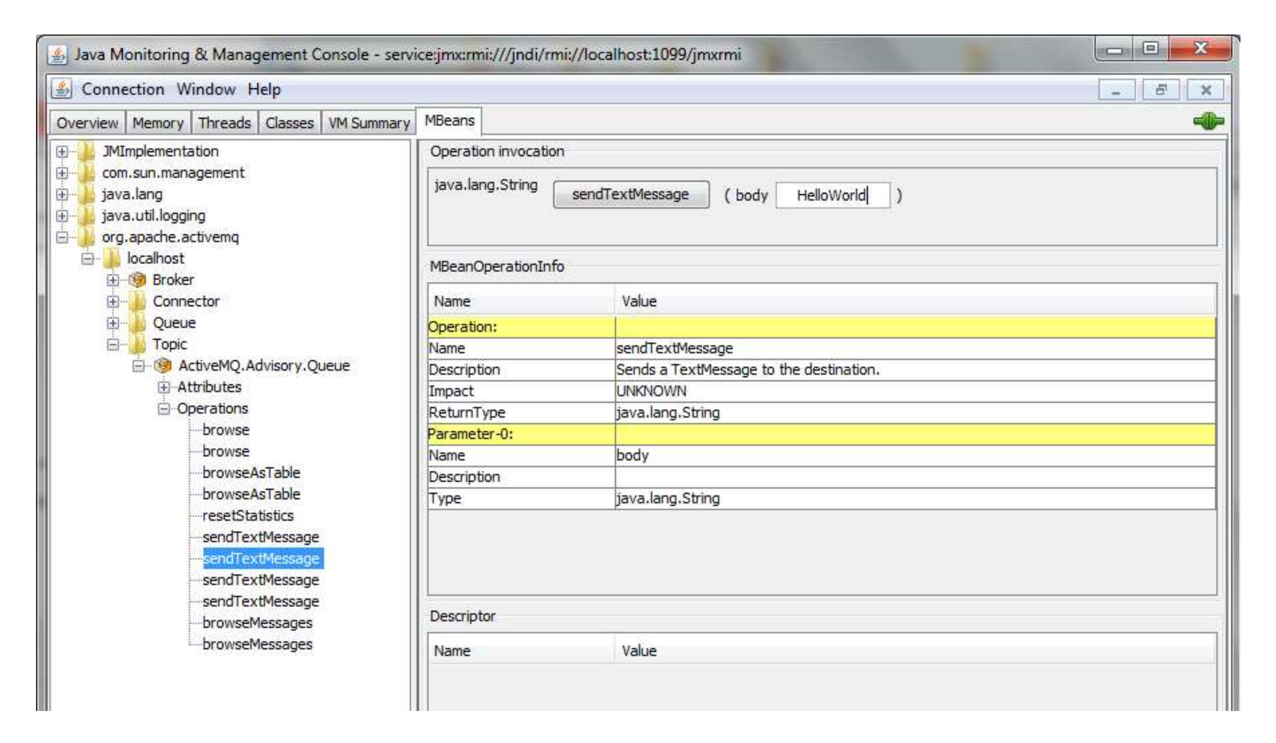


Choisir Remote Process et entrer le texte suivant :
service:jmx:rmi:///jndi/rmi://localhost:1099/jmxrmi , cliquer sur Connect :





Aller vers org.apache.activemq->localhost->Topic->SampleSubscriptionTopic->Operations->sendTextMessage :



Dans la zone de texte à côté du bouton SendTextMessage, entrer un texte à envoyer au Topic.

Le texte envoyé apparaît dans la console de Visual Studio.

Le producteur de message peut aussi être crée sous Visual Studio comme étant un nouveau projet et qui envoie le message vers le provider ActiveMQ.